

Raft Warfighting Data Model (NGC2)

Contents

| | |
|---|----------|
| Raft Next Generation Command and Control (NGC2) Warfighting Data Model (WDM) | 3 |
| Protocol Documentation | 4 |
| Entity | 4 |
| AviationInfo | 4 |
| Circle | 4 |
| CovarianceMatrix3 | 4 |
| Ellipse | 5 |
| EmitterNotation | 5 |
| Entity | 5 |
| Flexible Field Model | 6 |
| Required vs Optional Fields | 6 |
| Examples by Entity Type | 6 |
| Field Population Guidelines | 7 |
| Entity.LabelsEntry | 9 |
| EntityAssessment | 9 |
| EntityFlags | 10 |
| EntityTypeInfo | 10 |
| ErrorEllipse | 10 |
| Expendables | 11 |
| FieldOfView | 11 |
| 3D Cone | 11 |
| 2D Projected Footprint | 11 |
| FuelState | 12 |
| GroundInfo | 12 |
| GunState | 13 |
| Identity | 13 |
| LineOfBearing | 13 |
| LinearRing | 13 |
| MaritimeInfo | 14 |

| | |
|-------------------------------|----|
| Mode5 | 14 |
| Motion | 14 |
| Ontology | 15 |
| OrbitalInfo | 15 |
| OrbitalMotion | 16 |
| Ordnance | 16 |
| Orientation | 16 |
| PhysicalDimensions | 17 |
| Point | 17 |
| Polygon | 17 |
| Polyline | 18 |
| Position | 18 |
| Rectangle | 19 |
| ScanBehavior | 19 |
| Sensor | 19 |
| SideModifiers | 20 |
| SignalInfo | 21 |
| SpatialGeometry | 21 |
| SpatialLocation | 22 |
| SpeedModifier | 22 |
| StateVector | 23 |
| Symbology | 23 |
| TargetingInfo | 23 |
| TrackInfo | 23 |
| TransponderCodes | 24 |
| Vector3 | 24 |
| CoverStatus | 24 |
| Disposition | 25 |
| EmergencyStatus | 25 |
| EmitterNotationType | 25 |
| EntityCategory | 26 |
| EntityStatus | 26 |
| Environment | 27 |
| IFFResponse | 27 |
| MovementStatus | 27 |
| NavigationStatus | 27 |
| PriorityTier | 28 |
| ScanType | 28 |
| SideVersion | 28 |
| SignalModulation | 29 |
| TerrainType | 29 |
| ThreatLevel | 29 |

| | |
|----------------------------------|----|
| VerticalMode | 30 |
| VesselType | 30 |
| WakeCategory | 30 |
| Provenance | 30 |
| Provenance | 30 |
| Reconciliation | 31 |
| ReconciliationMetadata | 31 |
| ReconciliationAction | 32 |
| Security | 32 |
| SecurityComponents | 32 |
| SecurityMarking | 32 |
| Task | 33 |
| Principal | 33 |
| Task | 33 |
| TaskProgress | 34 |
| PrincipalType | 34 |
| TaskPriority | 34 |
| TaskState | 35 |
| Scalar Value Types | 35 |

Raft Next Generation Command and Control (NGC2) Warfighting Data Model (WDM)

The Raft Warfighting Data Model (WDM) is a structured data model for tactical command and control, composed for the U.S. Army’s Next Generation Command and Control (NGC2) program. It provides a shared language for capturing the operational picture: what exists in the battlespace, and what is being done about it.

The model is designed for explicit interoperability. External identifiers and provenance metadata allow WDM records to bridge to other tactical data models and source systems, including OMS UCI messages, Cursor on Target (CoT) events, AIS, ADS-B, Link-16 track reports, and GCCS-J.

Ontology alignment with BFO and the Common Core Ontologies (CCO) provides a formal semantic foundation, enabling WDM records to participate in cross-domain knowledge representation and reasoning alongside other DoW and IC ontology-aligned systems.

The Raft Data Platform (RDP) provides the runtime for this model. RDP offers a federated data catalog, eventually-consistent mesh replication, data governance and distribution policies, and related platform services. WDM defines the structure, while RDP handles storage, transport, and access control. Every record carries security markings, provenance, and replication metadata so that data can be trusted, attributed, and shared across classification boundaries and disconnected networks.

Protocol Documentation

Entity

AviationInfo

Aviation domain attributes. From ADS-B, flight plans, radar, Link-16.

| Field | Type | Description |
|---------------|-----------------|--|
| aircraft_type | string | Aircraft type (e.g., “F-35A Lightning II”, “C-17A Globemaster III”). |
| icao_type | string | ICAO type designator (e.g., “B738”, “F35”). |
| wake_category | WakeCategory | Wake turbulence category. |
| registration | string | Tail number / registration. |
| operator | string | Operating unit, command, or organization. |
| flight_number | string | Callsign or flight number (e.g., “REACH 421”, “EVAC 01”). |
| origin | string | Departure airport (ICAO code). |
| destination | string | Destination airport (ICAO code). |
| emergency | EmergencyStatus | Emergency status. |
| vertical_mode | VerticalMode | Vertical flight mode. |

Circle

A circle defined by a center point and radius.

No GeoJSON equivalent but widely used in military applications.

Examples: engagement area, weapon effects radius, sensor range ring, circular error probable (CEP).

| Field | Type | Description |
|---------------|----------|---|
| center | Position | Center position of the circle. |
| radius_meters | double | Radius in meters. Must be positive. |
| height_meters | double | Optional height above center to extrude in meters. A non-zero value creates a cylinder. |

CovarianceMatrix3

Upper triangle of a symmetric 3x3 matrix.

For ENU covariance: xx=East-East, yy=North-North, zz=Up-Up.

[xx xy xz] [yy yz] [zz]

| Field | Type | Description |
|-------|--------|-------------|
| xx | double | |

| Field | Type | Description |
|-------|--------|-------------|
| xy | double | |
| xz | double | |
| yy | double | |
| yz | double | |
| zz | double | |

Ellipse

An ellipse defined by center, semi-axes, and orientation.

For a circle, set semi_major_axis_meters == semi_minor_axis_meters. Not GeoJSON compatible.

Examples: positional uncertainty ellipse, radar coverage footprint, communications coverage area.

| Field | Type | Description |
|------------------------|----------|--|
| center | Position | Center position of the ellipse. |
| semi_major_axis_meters | double | Semi-major axis length in meters (the longer radius). |
| semi_minor_axis_meters | double | Semi-minor axis length in meters (the shorter radius). |
| orientation_degrees | double | Orientation of the semi-major axis in degrees clockwise from true north. Range: 0.0 to 180.0 (symmetric across the semi-minor axis). |
| height_meters | double | Optional height above center to extrude in meters. A non-zero value creates an elliptic cylinder. |

EmitterNotation

Emitter notation identification (ELNOT/CENOT).

| Field | Type | Description |
|------------|---------------------|---|
| notation | string | ELNOT or CENOT identifier string. |
| confidence | optional double | Confidence that this identification is correct (0.0-1.0). |
| type | EmitterNotationType | Whether this notation is an ELNOT or CENOT. |

Entity

Core representation of a tracked object in the battlespace.

An Entity is anything with identity that warfighters need to track: platforms, units, facilities, equipment, personnel, events, control measures, etc.

Flexible Field Model

Entities use a **field-optional** design rather than strict entity subtypes. Most fields are optional - populate only what is relevant for the specific entity being represented. This flexibility allows the schema to represent diverse entity types without requiring separate message definitions for each category.

Different entity types naturally use different field subsets: - **Maritime vessel**: location, motion, maritime info, identities (MMSI), dimensions - **Aircraft**: location, motion, aviation info, identities (Mode-S), assessment - **Ground vehicle**: location, motion, ground info, assessment, labels - **Facility**: location, dimensions, assessment (no motion) - **Person**: location, assessment (minimal fields) - **Control measure**: shape (or geometry), assessment, labels (no motion/dimensions) - **Event**: location, provenance, description (transient, no motion)

Required vs Optional Fields

Minimally required for all entities: - `id`: Unique identifier (generated if not provided) - `provenance.updated_at`: When this data was last modified

Commonly populated but optional: - `name`: Human-readable identifier - `location.position`: Geographic coordinates - `assessment`: Tactical assessment (disposition, environment)

Domain-specific (populate as applicable): - `type_info.maritime`: For ships, boats, submarines - `type_info.aviation`: For aircraft, helicopters, UAVs - `type_info.ground`: For vehicles, dismounts, equipment - `type_info.orbital`: For satellites, space objects - `type_info.signal`: For SIGINT/ELINT emissions

Motion-related (omit for stationary entities): - `motion`: Velocity and acceleration (platforms in motion) - `orbital_motion`: Keplerian elements (satellites) - `state_vector`: Position and velocity vectors (high-precision orbital)

Specialized use cases: - `shape`: Structured geometry (polygons, ellipses, circles, polylines) - `geometry`: Extended spatial representation as WKT string - `targeting`: When entity is a potential target - `dimensions`: Physical measurements when known - `details`: System-specific extensions

Examples by Entity Type

Naval vessel (comprehensive):

```
{
  "id": "...", "name": "USS Arleigh Burke", "location": {...}, "motion": {...},
  "type_info": {"maritime": {"vessel_type": "MILITARY", "mmsi": "..."}},
  "assessment": {"disposition": "FRIEND", "environment": "SURFACE"},
  "dimensions": {...}, "provenance": {...}
}
```

Aircraft (typical):

```
{
  "id": "...", "name": "REACH 421", "location": {...}, "motion": {...},
  "type_info": {"aviation": {"aircraft_type": "C-17A", "callsign": "REACH421"}},
  "assessment": {"disposition": "FRIEND"}, "provenance": {...}
}
```

Ground vehicle (minimal):

```
{
  "id": "...", "location": {...}, "motion": {...},
  "assessment": {"disposition": "HOSTILE", "environment": "SURFACE"},
  "provenance": {...}
}
```

Facility (no motion):

```
{
  "id": "...", "name": "FOB Lightning", "location": {...},
  "assessment": {"disposition": "FRIEND", "dimensions": {...},
  "provenance": {...}
}
```

Event (transient):

```
{
  "id": "...", "name": "IED Strike", "description": "Vehicle-borne IED...",
  "location": {...}, "provenance": {...}
}
```

Field Population Guidelines

1. **Populate what you know** - Omit unknown or irrelevant fields
2. **Don't invent data** - Empty/default values can mislead consumers
3. **Use type_info discriminator** - Populate the domain-specific section that applies
4. **Provenance is critical** - Always include source and timestamp
5. **Assessment is required for warfighting context** - Disposition and environment provide tactical context

The authoritative timestamp for entity updates is in `provenance.updated_at`.

| Field | Type | Description |
|-------------|-------------------|---|
| id | string | Unique identifier (UUID). |
| security | SecurityMarking | Classification and handling caveats. |
| name | string | Name given to this entity. |
| description | string | Human-readable description or remarks. |
| identities | repeated Identity | External identifiers from source systems. Examples: MMSI, Mode-S address, Link-16 track number. |
| flags | EntityFlags | Operational flags (simulated, exercise). |
| type_info | EntityTypeInfo | Domain-specific attributes (maritime, aviation, ground). |
| provenance | Provenance | Data lineage and source attribution. |
| location | SpatialLocation | Geographic position, orientation, and uncertainty. |
| motion | Motion | Velocity and acceleration. |

| Field | Type | Description |
|----------------|-----------------------------|--|
| geometry | string | Extended geometry for non-point entities (WKT format). Use for unit boundaries, routes, engagement areas. Examples: POINT(lon lat) - "POINT(69.2 34.5)" (Kabul) LINESTRING(lon lat, ...) - "LINESTRING(44.3 33.3, 44.4 33.4)" (MSR Tampa) POLYGON((lon lat, ...)) - "POLYGON((44 33, 45 33, 45 34, 44 34, 44 33))" (AO) CIRCLE((lon lat), radius_meters) - Non-standard but commonly supported |
| assessment | EntityAssessment | Warfighter's assessment: affiliation, environment, confidence. This is ASSESSED information, not ground truth. |
| dimensions | PhysicalDimensions | Physical size and weight. |
| targeting | TargetingInfo | Targeting data if this entity is a potential target. |
| labels | repeated Entity.LabelsEntry | Extensible key-value metadata. Keys should use namespace convention: "domain.system.field" |
| orbital_motion | OrbitalMotion | Motion for orbiting space objects. |
| state_vector | StateVector | State vector for orbiting space objects. |
| ontology | Ontology | Semantic classification and formal ontology alignment. Includes operational category and optional references to external ontologies (BFO, CCO, domain-specific) for interoperability. |
| details | google.protobuf.Struct | Entity-specific metadata as structured data. Use this field when entity information isn't represented in the base Entity fields. Clients can include domain-specific data (e.g., fire unit ammunition levels, target formation details, custom sensor attributes). Prefer strongly-typed fields (labels, type_info) when available for better schema enforcement. TYPE IDENTIFIER CONVENTION: For structured data, include an "@type" field to identify the schema type. This helps clients determine which protobuf message to unmarshal the data into. Type URL formats: - WDM extensions: "raft.wdm.ngc2.v1.ext.{MessageName}" See raft/wdm/ngc2/v1/ext for available extension types. - Custom types: Use reverse-DNS format (e.g., "com.acme.CustomType") - No @type: Generic/unstructured data where clients inspect fields directly Examples: WDM extension: { "@type": "raft.wdm.ngc2.v1.ext.FireUnit", "value": { "fs_system_type": ["FIRE_SUPPORT_SYSTEM_TYPE_FA_CANNON"], "echelon": "PLATOON" } } Custom type: { "@type": "com.acme.CustomSensorData", "value": { "sensor_id": "...", "readings": [...] } } No type identifier: { "notes": "Some custom data", "metadata": {...} } |
| status | EntityStatus | Current lifecycle status of this entity. Controls whether the entity is actively tracked or disabled. |
| ttl | google.protobuf.Timestamp | Time-to-live: when set, entity expires at this time. |
| persist | bool | Persist indefinitely: when true, entity never auto-expires. |
| symbology | Symbology | Military symbology information (MIL-STD-2525C/D). Provides the raw SIDC string plus typed, structured access to all 2525C symbol modifiers. Populate when symbology data is available. Applicable to all entity types including tactical graphics, FSCMs, and control measures that carry SIDCs without requiring an assessment. |
| shape | SpatialGeometry | Structured geometric shape for spatial representation. Typed alternative to the WKT geometry string field for programmatic access to shape data without string parsing. Populate either this field or geometry, not both. For simple point entities, use location instead. Examples: area of operations boundary, route, engagement area, uncertainty ellipse, sensor coverage footprint. |

| Field | Type | Description |
|-------------------|------------------------|---|
| reconciliation | ReconciliationMetadata | Reconciliation engine decision metadata. Read-only; populated by the reconciliation engine on every upsert. Tracks which policy made the decision, what action was taken, and why. Replicates with the entity through the mesh, giving receiving nodes context on how the entity was reconciled at the originating node. |
| track_info | TrackInfo | Track quality and measurement metadata. Provides consumers with information about how this entity's position was derived and how much to trust it. Distinct from EntityAssessment, which captures the warfighter's tactical assessment (disposition, environment, confidence in identity). |
| transponder_codes | TransponderCodes | IFF/transponder interrogation codes. Mode 1-5 and Mode S codes from IFF interrogation. Applicable to air, surface, and ground entities with transponders. |
| priority | PriorityTier | QoS priority tier assigned by the platform. Used by the mesh for routing and replication prioritization. Read-only; assigned by the platform. Any client-provided value is ignored. |
| correlation | CorrelationState | Correlation state linking this entity to a correlation set. A correlation set groups entities representing the same real-world object. Each entity carries its own set_id and role. There are no cross-entity references. Read-only on this message; managed exclusively via the dedicated correlation API, which provides atomic multi-entity updates. Any value provided on an entity upsert is ignored. When unpopulated, the entity is uncorrelated (standalone track). |
| decorrelations | repeated Decorrelation | Decorrelation records for entities explicitly excluded from correlation with this entity. An entity can be correlated (via correlation field) and decorrelated (via this list) simultaneously, as long as the targets are disjoint. Automated algorithms must check this list before proposing re-correlation. Manual decorrelations block automated re-correlation of the same pair. Capped at 25 records. The correlation service rejects a decorrelation request that would exceed this limit; callers must evict stale records before adding new ones. Read-only on this message; managed exclusively via the dedicated correlation API. Any value provided on an entity upsert is ignored. |
| sensors | repeated Sensor | Sensors mounted on this entity. Each entry groups one or more fields of view under a named sensor. Populate for entities carrying directed sensors (cameras, radars, sonars) where the coverage area is operationally relevant. |
| expendables | Expendables | Fuel, ordnance, and other expendables carried by the entity. Domain-agnostic: applicable to aircraft, ground vehicles, ships, and any other entity that carries fuel or munitions. |

Entity.LabelsEntry

| Field | Type | Description |
|-------|--------|-------------|
| key | string | |
| value | string | |

EntityAssessment

Warfighter's tactical assessment of this entity.

IMPORTANT: This is ASSESSED affiliation, not ground truth. The actual identity may differ. Drives MIL-STD-2525 symbology and ROE.

| Field | Type | Description |
|------------------|-----------------|--|
| disposition | Disposition | Friend/hostile/neutral/unknown determination. |
| environment | Environment | Operating domain (air, surface, subsurface, etc.). |
| nationality | string | Country of allegiance (ISO 3166-1 alpha-3: USA, GBR, CHN). |
| confidence_level | optional double | Confidence in this assessment (0.0 to 1.0). |

EntityFlags

Operational flags for data segregation.

| Field | Type | Description |
|--------------|------|---|
| is_simulated | bool | True if entity is computer-generated (not from real sensors). |
| is_exercise | bool | True if entity is part of an exercise (real or simulated). |
| is_taskable | bool | True if entity can receive tasks. |

EntityTypeInfo

Domain-specific attributes. Populate the relevant component based on entity domain.

| Field | Type | Description |
|----------|--------------|-------------|
| maritime | MaritimeInfo | |
| aviation | AviationInfo | |
| ground | GroundInfo | |
| orbital | OrbitalInfo | |
| signal | SignalInfo | |

ErrorEllipse

Uncertainty expressed as an error ellipse.

Captures sensor-reported error bounds without requiring conversion to a covariance matrix.

| Field | Type | Description |
|------------------------|-----------------|--|
| semi_major_axis_meters | optional double | Semi-major axis length in meters (the longer radius). |
| semi_minor_axis_meters | optional double | Semi-minor axis length in meters (the shorter radius). |
| orientation_degrees | optional double | Orientation of the semi-major axis in degrees clockwise from true north. Range: 0.0 to 180.0 (symmetric across the semi-minor axis). |

| Field | Type | Description |
|------------|-----------------|---|
| confidence | optional double | Probability that the true value lies within this ellipse. Range: 0.0 to 1.0. Omit when the confidence level is unknown. |

Expendables

Fuel, ordnance, and gun ammunition state.

Domain-agnostic: applicable to aircraft, ground vehicles, ships, and any other entity that carries fuel or munitions.

| Field | Type | Description |
|----------|--------------------|--|
| fuel | repeated FuelState | Current fuel state. Multiple entries for platforms with more than one fuel type (e.g., JP-8 + diesel on a naval vessel). |
| ordnance | repeated Ordnance | Expendable ordnance carried by the entity. Weapons, countermeasures, and other deployable stores. |
| guns | repeated GunState | Onboard gun ammunition state. Multiple entries for platforms with more than one gun system. |

FieldOfView

Field of view (FOV) for a sensor.

Describes a 3D viewing volume defined as an angular cone with optional range bounds. Supports both symmetric (circular) and asymmetric (rectangular) apertures.

The cone is oriented relative to the entity's body frame using an Orientation message (heading, pitch, roll). When the parent entity's location.orientation is also populated, consumers can compose the two to derive the absolute look direction in world coordinates.

3D Cone

The 3D viewing volume is defined by: - look_orientation: direction the boresight points (body-frame) - horizontal_fov_degrees: total azimuth span of the cone - vertical_fov_degrees: total elevation span of the cone - min_range_meters / max_range_meters: near and far range limits

For a symmetric (circular) FOV, set horizontal and vertical spans equal.

2D Projected Footprint

The ground_footprint field provides a pre-computed 2D projection of the viewing volume onto the terrain surface. This is a Polygon because terrain intersection of a cone generally produces a quadrilateral or irregular shape, not a simple circle or ellipse.

Producers that have access to a terrain model should populate this field. Consumers without terrain data can use it directly for map rendering without recomputing the projection.

| Field | Type | Description |
|------------------------|-----------------|---|
| look_orientation | Orientation | Boresight direction in the entity's body frame. heading_degrees: azimuth offset from the entity's nose (0 = forward, 90 = starboard). Range: 0.0 to 360.0. pitch_degrees: elevation offset from the entity's horizontal plane. Negative = looking down, positive = looking up. Range: -90.0 to +90.0. roll_degrees: rotation about the boresight axis. Range: -180.0 to +180.0. |
| horizontal_fov_degrees | optional double | Total horizontal (azimuth) angular extent of the FOV in degrees. The cone spans +/- half this value from the boresight azimuth. Range: >0.0 to 360.0. Example: 12.5 for a narrow EO camera, 120.0 for a wide-angle sensor. |
| vertical_fov_degrees | optional double | Total vertical (elevation) angular extent of the FOV in degrees. The cone spans +/- half this value from the boresight elevation. Range: >0.0 to 180.0. Example: 9.4 for a narrow EO camera. |
| min_range_meters | optional double | Minimum range in meters. Objects closer than this are outside the useful range. When unset, there is no minimum range constraint. |
| max_range_meters | optional double | Maximum range in meters. Objects beyond this are outside the useful range. When unset, range is unbounded or unknown. |
| ground_footprint | Polygon | Pre-computed 2D ground projection of the viewing volume. The polygon represents the intersection of the 3D viewing cone with the terrain surface. Producers with terrain models should populate this to spare consumers from recomputing the projection. Typically a quadrilateral (4 vertices + closing point) but may have more vertices when terrain is uneven. |

FuelState

Current fuel state for a single fuel type.

| Field | Type | Description |
|-------------|-----------------|---|
| fuel_type | string | Fuel type or designation (e.g., "JP-8", "F-76", "AVGAS"). |
| quantity_kg | optional double | Current fuel quantity in kilograms. |
| capacity_kg | optional double | Maximum fuel capacity in kilograms. |

GroundInfo

Ground domain attributes.

| Field | Type | Description |
|---------------|----------------|--|
| platform_type | string | Platform type (e.g., "T-72", "HMMWV", "dismount"). |
| movement | MovementStatus | Current movement state. |
| terrain | TerrainType | Terrain type at entity location. |
| cover | CoverStatus | Cover and concealment status. |

GunState

Onboard gun ammunition state.

| Field | Type | Description |
|------------------|--------|--|
| system | string | Gun system identifier (e.g., “M61A2”, “GAU-8/A”, “Mk 45 Mod 4”). |
| round_type | string | Type of rounds loaded (e.g., “20MM PGU-28”, “30MM PGU-13 HEI”). |
| rounds_remaining | uint32 | Rounds remaining. |

Identity

External system identifier. Enables correlation across systems (AIS, ADS-B, Link-16, etc.).

| Field | Type | Description |
|------------|--------|---|
| system | string | Source system name (e.g., “ais”, “adsb”, “link16”, “gccs-j”). |
| identifier | string | Identifier value from that system. |

LineOfBearing

| Field | Type | Description |
|-------------------------|-----------------|--|
| azimuth_degrees | optional double | Azimuth in degrees from True North. Range: 0.0 to 360.0 |
| azimuth_sigma_degrees | optional double | Uncertainty (1 Standard Deviation) in degrees. |
| elevation_degrees | optional double | Elevation angle in degrees from the horizontal plane. Positive values indicate above horizon, negative below. Range: -90.0 to 90.0 |
| elevation_sigma_degrees | optional double | Elevation uncertainty (1 Standard Deviation) in degrees. |
| distance_meters | optional double | Estimated distance to the signal source in meters. |
| distance_sigma_meters | optional double | Distance uncertainty (1 Standard Deviation) in meters. |

LinearRing

A closed ring of positions forming a polygon boundary.

The first and last positions must be identical to close the ring. Must contain at least 4 positions (3 distinct vertices + closing point).

Winding order follows GeoJSON convention: - Exterior rings: counter-clockwise. - Interior rings (holes): clockwise.

| Field | Type | Description |
|-----------|-------------------|--|
| positions | repeated Position | Ordered positions forming the closed ring. |

MaritimeInfo

Maritime domain attributes. Primarily from AIS.

| Field | Type | Description |
|-------------------|------------------|--|
| vessel_type | VesselType | Vessel classification. |
| imo_number | string | IMO ship identification number (7 digits). |
| mmsi | string | Maritime Mobile Service Identity (9 digits). |
| maritime_callsign | string | Radio callsign. |
| flag_state | string | Flag state (ISO 3166-1 alpha-3). |
| destination | string | Destination port or area. |
| pennant_number | string | Pennant number (pendant number) (hull number) for naval vessels. Official identifier assigned by naval authorities (e.g., “DDG-51”, “CVN-68”). |
| nav_status | NavigationStatus | AIS navigation status. |
| draft_meters | optional double | Current draft in meters. |
| cargo_type | string | Cargo type description. |
| sconum | string | Ship Control Number (SCONUM) see references: https://apps.dtic.mil/sti/tr/pdf/ADA562336.pdf https://niem.github.io/model/4.0/m/VesselSCONUMText/index.html |

Mode5

Mode 5 IFF interrogation data.

| Field | Type | Description |
|-------------|-----------------|---------------------------------|
| response | IFFResponse | Interrogation response status. |
| code | optional uint32 | Mode 5 code. |
| platform_id | optional uint32 | Platform identification number. |

Motion

Velocity and acceleration in the local East-North-Up (ENU) frame.

ENU is a local tangent plane coordinate system: East: positive toward geographic east North: positive toward geographic north Up: positive away from Earth center (perpendicular to ellipsoid)

| Field | Type | Description |
|-----------------------|-----------------|--|
| velocity_enu_mps | Vector3 | Velocity components (m/s). x=East, y=North, z=Up (positive = climbing). |
| acceleration_enu_mps2 | Vector3 | Acceleration components (m/s ²). Non-zero values indicate maneuvering. |
| speed_mps | optional double | Speed over ground (m/s). Horizontal magnitude: $\sqrt{\text{east}^2 + \text{north}^2}$. |

| Field | Type | Description |
|---------------------|-------------------|---|
| velocity_covariance | CovarianceMatrix3 | Velocity covariance in the local ENU frame (m/s) ² . Upper triangle of the 3x3 symmetric covariance matrix. Indicates how uncertain the velocity estimate is; used for dead-reckoning quality and distinguishing stable from noisy tracks. |

Ontology

Semantic classification of the entity.

Combines operational typing with extensible ontology references. The `category` enum provides a fixed operational classification grounded in BFO (Basic Formal Ontology) and CCO (Common Core Ontologies), while `refs` allows ontologists to supply formal type URIs from any ontology (BFO, CCO, DICO, domain-specific, etc.) without schema changes.

This supports DNI/DIA mandates for BFO/CCO alignment while remaining ontology-agnostic at the schema level.

CCO Alignment Notes: - `Entity.name` maps to `cco:designated_by` -> `cco:DesignativeName` - `Entity.identities` maps to `cco:designated_by` -> `cco:CodeIdentifier` - `Entity.description` maps to `cco:described_by` -> `cco:DescriptiveInformationContentEntity` - `Entity.location` maps to `bfo:located_in` -> `cco:GeospatialRegion`

Example ontology_refs: “bfo:IndependentContinuant” “cco:Aircraft” “cco:MilitaryOrganization” “mil:APP-6D/SFGPUCIZ—”

| Field | Type | Description |
|----------|-----------------|---|
| type | string | Specific type name (platform designation or unit type). The authoritative designator for this entity type per DOD/NATO standards. Examples: “F-35A Lightning II”, “DDG-51 Arleigh Burke”, “T-72B3”, “Infantry Battalion”, “SA-21 Growler” |
| category | EntityCategory | Operational entity category. What kind of thing is this from a warfighting perspective? Each category is grounded in BFO/CCO upper ontology. |
| refs | repeated string | Formal ontology type references (URIs or CURIEs). Allows ontologists to map entities to external ontologies such as BFO, CCO, DICO, or domain-specific vocabularies. Format: prefix:LocalName or full URI Common prefixes: bfo: http://purl.obolibrary.org/obo/ cco: http://www.ontologyrepository.com/CommonCoreOntologies/ mil: (MIL-STD-2525D symbol codes) Examples: “bfo:IndependentContinuant” “cco:GroundVehicle” “cco:MilitaryOrganization” |

OrbitalInfo

| Field | Type | Description |
|-------------|---------------------------|---|
| object_id | string | The ID of the on-orbit object (ex. satellite number) |
| sat_cat | string | Satellite Catalog number, if applicable. |
| object_type | string | The type of object (active satellite, inactive satellite, debris, natural object) |
| operator | string | Current operator of this orbital object, if applicable. |
| origin | string | Country or organization of origin, if known. |
| launch_date | google.protobuf.Timestamp | Timestamp when this object was launched. |

OrbitalMotion

Motion fields that generally match a Two-Line Element set.

| Field | Type | Description |
|---------------|---------------------------|--|
| epoch | google.protobuf.Timestamp | Epoch of this kinematic measurement. |
| mean_mot_dot | double | First derivative of mean motion. |
| mean_mot_ddot | double | Second derivative of mean motion. |
| b_star | double | Drag/radiation pressure coefficient. |
| inclination | double | Orbital inclination in degrees. Range: 0.0 to 180.0 0° = equatorial, 90° = polar, >90° = retrograde Angle between orbital plane and Earth's equatorial plane. |
| raan | double | Right Ascension of Ascending Node (RAAN) in degrees. Range: 0.0 to 360.0 Celestial longitude where orbit crosses equatorial plane northbound. Measured eastward from vernal equinox. |
| eccentricity | double | Orbital eccentricity (dimensionless). Range: 0.0 (circular) to <1.0 (elliptical) Defines the shape of the orbit. 0 = perfect circle, approaching 1 = highly elliptical. |
| arg_perigee | double | Argument of perigee in degrees. Range: 0.0 to 360.0 Angle from ascending node to perigee (closest approach point). Defines orbital ellipse orientation within the orbital plane. |
| mean_anomaly | double | Mean anomaly in degrees. Range: 0.0 to 360.0 Angular position of satellite along orbit at epoch time. 0° = perigee, 180° = apogee. |
| rev_num | double | Revolution number at epoch. Total number of complete orbits since launch. Increments by 1 each time satellite crosses ascending node. |

Ordnance

A single expendable ordnance type carried by the entity. Covers weapons (missiles, bombs, torpedoes), countermeasures (flares, chaff, decoys), and other deployable stores (sonobuoys).

| Field | Type | Description |
|----------|--------|--|
| type | string | Designator (e.g., "AIM-120C", "MJU-27", "SSQ-62"). |
| name | string | Human-readable name (e.g., "AMRAAM", "Flare", "Sonobuoy"). |
| quantity | uint32 | Number of units currently carried. |

Orientation

Orientation (attitude) in 3D space using Euler angles.

Convention: Tait-Bryan angles (aerospace sequence). Order: Yaw (heading) -> Pitch -> Roll

| Field | Type | Description |
|-----------------|-----------------|---|
| heading_degrees | optional double | Heading (yaw) in degrees from true north. Range: 0.0 to 360.0 (clockwise from north) Aircraft: direction nose is pointing Ships: direction bow is pointing |

| Field | Type | Description |
|---------------|-----------------|--|
| pitch_degrees | optional double | Pitch angle in degrees. Positive: nose/bow up Negative: nose/bow down Range: -90.0 to +90.0 |
| roll_degrees | optional double | Roll angle in degrees. Positive: right wing/starboard down Negative: left wing/port down Range: -180.0 to +180.0 |

PhysicalDimensions

Physical size and mass of the entity. Used for recognition, classification, and logistics planning.

| Field | Type | Description |
|---------------|-----------------|--|
| length_meters | optional double | Overall length (bow to stern, nose to tail). |
| width_meters | optional double | Overall width (beam for ships, wingspan for aircraft). |
| height_meters | optional double | Overall height (keel to mast, wheels to tail). |
| weight_kg | optional double | Mass (displacement for ships, MTOW for aircraft). |

Point

A single geographic point.

GeoJSON equivalent: Point. See <https://datatracker.ietf.org/doc/html/rfc7946#section-3.1.2>

Examples: observation post, waypoint, point target.

| Field | Type | Description |
|----------|----------|-----------------------------------|
| position | Position | Geographic position of the point. |

Polygon

A closed polygon defined by an exterior ring and optional interior holes.

GeoJSON equivalent: Polygon (only canonical representations accepted). See <https://datatracker.ietf.org/doc/html/rfc7946#section-3.1.6>

Follows the right-hand rule: exterior rings are counter-clockwise, interior rings (holes) are clockwise.

Examples: area of operations (AO), engagement area, named area of interest (NAI), restricted operations zone (ROZ).

| Field | Type | Description |
|-------|---------------------|--|
| rings | repeated LinearRing | Rings defining the polygon boundary. rings[0]: exterior boundary (required). rings[1..N]: interior holes (optional). |

| Field | Type | Description |
|---------------|--------|---|
| height_meters | double | Optional uniform height above base positions to extrude in meters. Creates a prism from the 2D polygon. When 0/unset, polygon is 2D. Strictly GeoJSON-compatible polygons will not have this set. |

Polyline

An ordered sequence of positions forming a line or path.

GeoJSON equivalent: LineString. See <https://datatracker.ietf.org/doc/html/rfc7946#section-3.1.4>

Must contain at least 2 positions.

Examples: route, main supply route (MSR), phase line, flight path.

| Field | Type | Description |
|-----------|-------------------|--|
| positions | repeated Position | Ordered sequence of positions defining the line. |

Position

Geographic position in WGS-84 coordinates.

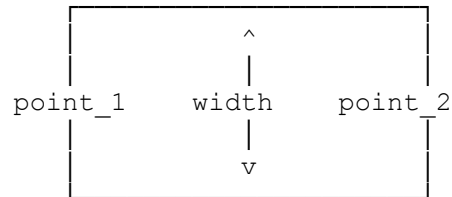
Supports multiple altitude references for interoperability: HAE: GPS-native, precision applications MSL: Aviation (flight levels, altimeters) AGL: Low-altitude ops, terrain following Depth: Subsurface operations

| Field | Type | Description |
|---------------------|-----------------|--|
| latitude_degrees | double | WGS-84 geodetic latitude in decimal degrees. Range: -90.0 (South Pole) to +90.0 (North Pole) Precision: 7 decimal places = ~1cm accuracy |
| longitude_degrees | double | WGS-84 geodetic longitude in decimal degrees. Range: -180.0 (West) to +180.0 (East) Precision: 7 decimal places = ~1cm at equator |
| altitude_hae_meters | optional double | Height Above Ellipsoid (HAE) in meters. WGS-84 reference ellipsoid. Positive above ellipsoid; negative below. Preferred for precision applications. |
| altitude_agl_meters | optional double | Altitude Above Ground Level (AGL) in meters. Requires terrain elevation database. Common for low-altitude flight operations. |
| altitude_msl_meters | optional double | Mean Sea Level (MSL) altitude in meters. Reference: geoid (mean sea level surface) Common in aviation (flight levels) but less precise. |
| depth_meters | optional double | Depth below surface in meters (for subsurface). Positive values = depth below surface. Derived from pressure; affected by water density. |
| mgrs | string | Military Grid Reference System coordinate. Format: Grid Zone + 100km Square + Easting + Northing Example: "4QFJ 12345 67890" (10-digit = 1m precision) |

Rectangle

A rectangle defined by two centerline points and a width.

The two points define the centerline (long axis) of the rectangle. The width extends perpendicular to that line on both sides. Orientation is implicit from the bearing of point_1 to point_2.



Structurally guarantees a valid rectangle.

Examples: kill box, fire support area, restricted fire area (RFA), no-fire area (NFA), free-fire area (FFA).

| Field | Type | Description |
|---------------|----------|--|
| point_1 | Position | First endpoint of the rectangle centerline. |
| point_2 | Position | Second endpoint of the rectangle centerline. |
| width_meters | double | Total width in meters, extending perpendicular to the centerline. Half the width extends to each side. Must be positive. |
| height_meters | double | Optional height above the centerline to extrude in meters. A non-zero value creates a rectangular prism. |

ScanBehavior

Scanning behavior of a signal source.

| Field | Type | Description |
|---------------------|-----------------|-------------------------|
| scan_type | ScanType | Type of scan pattern. |
| scan_period_seconds | optional double | Scan period in seconds. |

Sensor

A sensor mounted on an entity.

Groups one or more fields of view under a named, typed sensor. Examples: “search radar”, “spherical sonar array”, “EO turret”.

| Field | Type | Description |
|----------------|----------------------|--|
| name | string | Human-readable name for this sensor. Examples: “search radar”, “spherical sonar array”, “EO turret”. |
| type | string | Sensor type or classification. Examples: “radar”, “sonar”, “eo”, “ir”, “lidar”, “sar”. |
| fields_of_view | repeated FieldOfView | Fields of view for this sensor. |

SideModifiers

Typed MIL-STD-2525C symbol modifier fields.

Each field corresponds to a modifier defined in MIL-STD-2525C, Appendix A, Table XIV. The field comment includes the 2525C modifier letter code in parentheses. All fields are optional; populate only what is known and relevant.

Reference: MIL-STD-2525C, Appendix A, Table XIV.

| Field | Type | Description |
|-----------------------|---------------------------|--|
| quantity | int32 | (C) Quantity of equipment or personnel represented by the symbol. |
| reinforced_reduced | string | (F) Reinforced or reduced status. Values: “R” (reinforced), “D” (reduced), “RD” (reinforced and reduced). |
| staff_comments | string | (G) Staff comments. Free-form text annotation (max 20 chars per spec). |
| additional_info | string | (H) Additional information. Free-form text (max 20 chars per spec). |
| additional_info_2 | string | (H1) Additional information, second line. |
| additional_info_3 | string | (H2) Additional information, third line. |
| evaluation_rating | string | (J) Evaluation rating. Reliability and credibility indicator (2-character code per STANAG 2022). |
| combat_effectiveness | string | (K) Combat effectiveness. Unit readiness or installation capability indicator (max 5 chars per spec). |
| signature_equipment | string | (L) Signature equipment. Detectable electronic signature. |
| higher_formation | string | (M) Higher formation. Parent unit or higher echelon designation (max 21 chars per spec). |
| iff_sif | string | (P) IFF/SIF identification mode and code (max 5 chars per spec). |
| direction_of_movement | double | (Q) Direction of movement in degrees from true north (0.0-360.0). |
| sigint_mobility | string | (R2) SIGINT mobility indicator code. |
| unique_designation | string | (T) Unique designation. Primary identifier for the symbol (max 21 chars per spec). |
| unique_designation_2 | string | (T1) Unique designation, second line. |
| type | string | (V) Type of equipment. Platform or equipment descriptor (max 24 chars per spec). |
| effective_time | google.protobuf.Timestamp | (W) Date-time group for effective time. Stored as a structured timestamp; render as DTG for display. |
| expiration_time | google.protobuf.Timestamp | (W1) Date-time group for expiration time. Stored as a structured timestamp; render as DTG for display. |
| altitude_depth | repeated double | (X) Altitude or depth values in meters. Supports multiple values for multi-point tactical graphics. |
| location | string | (Y) Location in any desired display format (max 19 chars per spec). Typically DMS, MGRS, or other coordinate representation. |
| speed | SpeedModifier | (Z) Speed of the symbol. Structured as value + unit for clarity. |

| Field | Type | Description |
|-------------------------|-----------------|--|
| special_c2_headquarters | string | (AA) Special C2 headquarters designation. |
| platform_type | string | (AD) Platform type. ELNOT or CENOT notation for SIGINT symbols. |
| equipment_teardown_time | double | (AE) Equipment teardown time in minutes. |
| common_identifier | string | (AF) Common identifier. Common name for the equipment (e.g., “Hawk” for a Hawk SAM system). |
| distance | repeated double | (AM) Distance in meters. Supports multiple values for radius, length, width of tactical graphics. |
| azimuth | repeated double | (AN) Azimuth values in degrees from true north. Supports multiple values for tactical graphics with multiple orientation parameters. |
| engagement_bar | string | (AO) Engagement bar. Target engagement status indicator. |

SignalInfo

Signal domain attributes.

Core fields (frequency, RSSI, line of bearing) are sufficient for basic signal representation. Extended fields (SNR, pulse characteristics, scan behavior, emitter notations) support ELINT classification and EW-to-fires candidate scoring.

| Field | Type | Description |
|---------------------|--------------------------|--|
| frequency_hz | optional double | Center frequency of the signal in Hz. |
| rssi_dbm | optional double | Received Signal Strength Indicator (RSSI) in dBm. |
| line_of_bearing | LineOfBearing | Measurement of direction towards the signal source. |
| snr_db | optional double | Signal-to-noise ratio in dB. |
| pulse_width_seconds | optional double | Pulse width in seconds. |
| pri_seconds | optional double | Pulse repetition interval in seconds. |
| bandwidth_hz | optional double | Signal bandwidth in Hz. |
| emitter_notations | repeated EmitterNotation | Emitter notation identifiers with classification confidence. ELNOT or CENOT identifiers that classify the emitter. |
| scan | ScanBehavior | Scanning behavior of the signal source. |
| modulation | SignalModulation | Modulation type of the signal. Makes the signal class explicit rather than requiring consumers to infer it from which pulse fields are populated. |
| frequency_min_hz | optional double | Minimum frequency in Hz for agile or hopping emitters. For fixed-frequency signals, this equals frequency_hz. Populate alongside frequency_hz (center) when the emitter hops or sweeps across a range. |
| frequency_max_hz | optional double | Maximum frequency in Hz for agile or hopping emitters. |

SpatialGeometry

Structured geometric shape for spatial representation.

Provides typed alternatives to the WKT-based Entity.geometry string field. Use for boundaries, coverage areas, routes, sensor footprints, and other non-point geometries.

Exactly one shape type must be set. The shape defines the spatial extent of an entity: boundaries, coverage areas, routes, sensor footprints, and other non-point geometries.

For simple point entities, use Entity.location instead.

GeoJSON compatibility: - Point, Polyline, and Polygon map directly to GeoJSON Point, LineString, and Polygon types. - Circle and Ellipse have no GeoJSON equivalent but are common in military and geospatial applications.

| Field | Type | Description |
|-----------|-----------|---|
| point | Point | Single geographic point. |
| polyline | Polyline | Ordered sequence of positions forming a line. |
| polygon | Polygon | Closed area defined by one or more rings (exterior boundary + holes). |
| circle | Circle | Circle defined by center point and radius. |
| ellipse | Ellipse | Ellipse defined by center, semi-axes, and orientation. |
| rectangle | Rectangle | Rectangle defined by two centerline points and a width. |

SpatialLocation

Geographic position and orientation of an entity.

Combines where something is (position) with how it's oriented (attitude). Position uses WGS-84 datum with multiple altitude representations to support different operational contexts.

| Field | Type | Description |
|------------------------|-------------------|---|
| position | Position | Geographic position (lat/lon/alt). |
| orientation | Orientation | Orientation (attitude) in 3D space. Aircraft: heading, pitch, roll. Ships: heading primarily. Ground vehicles: heading. |
| position_covariance | CovarianceMatrix3 | Position covariance in the local ENU frame (meters ²). Upper triangle of the 3x3 symmetric covariance matrix. Enables consumers to reason about position accuracy for engagement zones, correlation gating, and fires nomination. |
| position_error_ellipse | ErrorEllipse | Source-reported positional error ellipse. Center is implied by the position field. |

SpeedModifier

Speed with explicit unit of measure.

Used for the MIL-STD-2525C speed modifier (Z) where the unit system may vary by operational context.

| Field | Type | Description |
|-------|--------|---|
| value | double | Numeric speed value. |
| unit | string | Unit of measure (e.g., "kt", "km/h", "m/s", "mph"). |

StateVector

| Field | Type | Description |
|--------------------|---------------------------|---|
| epoch | google.protobuf.Timestamp | Epoch of this state vector. |
| reference_frame | string | Ex. J2000. |
| position_km | Vector3 | X, Y, and Z position in kilometers, in the given reference frame. |
| velocity_kmps | Vector3 | X, Y, and Z velocity in km/2. |
| acceleration_kmps2 | Vector3 | X, Y, and Z acceleration in km/s ² . |

Symbology

Typed military symbology representation supporting MIL-STD-2525C/D.

Provides both a raw SIDC string for interoperability and a decomposed, typed modifier structure for programmatic access. Consumers should prefer the typed modifiers when available; the raw sidc field serves as a canonical interchange format.

Example (2525C friendly ground mechanized infantry): symbology { sidc: "SFGPUCIZ-*****" version: SIDC_VERSION_2525C modifiers { unique_designation: "1-87 IN" higher_formation: "3BCT" combat_effectiveness: "GREEN" } }

| Field | Type | Description |
|-----------|---------------|--|
| sidc | string | Raw Symbol Identification Code (SIDC) string. For 2525C: 15-character alphanumeric code. For 2525D: 20-character numeric code. |
| version | SideVersion | Version of the symbology standard this SIDC conforms to. |
| modifiers | SideModifiers | Typed 2525C modifier fields. Provides structured access to all standard symbol modifiers with appropriate data types for each field. |

TargetingInfo

Targeting and threat assessment data.

| Field | Type | Description |
|--------------|---------------|----------------------------------|
| is_hvt | optional bool | Designated as High Value Target. |
| threat_level | ThreatLevel | Assessed threat level. |

TrackInfo

Track quality and measurement metadata.

Provides consumers with information about how this entity's position was derived and how much to trust it. Distinct from EntityAssessment, which captures the warfighter's tactical assessment (disposition, environment, confidence in identity).

| Field | Type | Description |
|--------------------------|---------------------------|--|
| sensor_hits | optional uint32 | Number of sensor detections that contributed to this track. Higher values generally indicate more reliable tracks. |
| last_measurement_time | google.protobuf.Timestamp | Timestamp of the most recent sensor measurement for this entity. Distinct from provenance.updated_at, which is when the source system last published. A track can be published (updated_at = now) based on a measurement from 30 seconds ago. |
| radar_cross_section_dbsm | optional double | Radar cross section in dBsm. Helps with classification and indicates detectability. |
| object_count | optional uint32 | Estimated number of objects represented by this entity. Used for formation tracking where a single track represents multiple objects (e.g., convoy, flight of aircraft). |
| track_quality | optional uint32 | Track Quality (TQ) Representing the accuracy of the track position see references: https://apps.dtic.mil/sti/tr/pdf/ADA404334.pdf http://www.dodccrp.org/events/2000_CCRTS/html/pdf_papers/Track_1/030.pdf |

TransponderCodes

IFF/transponder interrogation data.

Mode 1-5 and Mode S codes from IFF interrogation. Applicable to air, surface, and ground entities with transponders.

| Field | Type | Description |
|-----------------|-----------------|---|
| mode_1 | optional uint32 | Mode 1 code (military mission code, 5-bit). |
| mode_2 | optional uint32 | Mode 2 code (military unit code, 12-bit). |
| mode_3a | optional uint32 | Mode 3/A code (ATC assigned, 12-bit octal). |
| mode_4_response | IFFResponse | Mode 4 interrogation response. |
| mode_5 | Mode5 | Mode 5 interrogation response and codes. |
| mode_s_address | optional uint32 | Mode S ICAO address (24-bit). |

Vector3

Generic 3D vector. For ENU velocity: x=East, y=North, z=Up. For ENU acceleration: same convention.

| Field | Type | Description |
|-------|--------|---|
| x | double | First component (East for ENU frame). |
| y | double | Second component (North for ENU frame). |
| z | double | Third component (Up for ENU frame). |

CoverStatus

Cover and concealment status.

| Name | Description |
|--------------------------|-------------------------|
| COVER_STATUS_UNSPECIFIED | |
| COVER_STATUS_EXPOSED | Fully visible, no cover |
| COVER_STATUS_PARTIAL | Some cover/concealment |
| COVER_STATUS_CONCEALED | Hidden from view |
| COVER_STATUS_DUG_IN | In prepared position |

Disposition

Standard Identity per MIL-STD-2525D.

| Name | Description |
|----------------------------|---------------------------------------|
| DISPOSITION_UNSPECIFIED | |
| DISPOSITION_PENDING | Awaiting identification |
| DISPOSITION_UNKNOWN | Cannot be determined |
| DISPOSITION_ASSUMED_FRIEND | Assumed friendly (unconfirmed) |
| DISPOSITION_FRIEND | Positively identified friendly |
| DISPOSITION_NEUTRAL | Neither friend nor threat |
| DISPOSITION_SUSPECT | Potentially hostile |
| DISPOSITION_HOSTILE | Declared hostile per ROE |
| DISPOSITION_JOKER | Friendly acting as hostile (exercise) |
| DISPOSITION_FAKER | Hostile acting as friendly (exercise) |

EmergencyStatus

Aircraft emergency status (from transponder).

| Name | Description |
|------------------------------|--------------|
| EMERGENCY_STATUS_UNSPECIFIED | |
| EMERGENCY_STATUS_NONE | |
| EMERGENCY_STATUS_GENERAL | Squawk 7700 |
| EMERGENCY_STATUS_MEDICAL | |
| EMERGENCY_STATUS_FUEL | Minimum fuel |
| EMERGENCY_STATUS_NO_COMM | Squawk 7600 |
| EMERGENCY_STATUS_HIJACK | Squawk 7500 |

EmitterNotationType

Emitter notation classification system.

| Name | Description |
|-----------------------------------|---|
| EMITTER_NOTATION_TYPE_UNSPECIFIED | |
| EMITTER_NOTATION_TYPE_ELNOT | Electronic Intelligence Notation – non-communications emitters (radars, navigation aids, electronic warfare systems). |
| EMITTER_NOTATION_TYPE_CENOT | Communications Emitter Notation – communications emitters (radios, data links, satellite uplinks). |

EntityCategory

Operational entity categories grounded in BFO/CCO upper ontology.

Each category maps to BFO (Basic Formal Ontology) and CCO (Common Core Ontologies) classes to ensure semantic interoperability across DOD, IC, and allied systems. Categories are designed for warfighting operations and align with Army FM 1-02, JP 3-0, and MIL-STD-2525D.

BFO Continuant vs Occurrent: - Most categories are Continuants (exist fully at any point in time) - EVENT and SIGNAL are Occurrents (unfold over time)

| Name | Description |
|---------------------------------|--|
| ENTITY_CATEGORY_UNSPECIFIED | |
| ENTITY_CATEGORY_PLATFORM | Individual vehicle, vessel, aircraft, or spacecraft. BFO: bfo:Object |
| ENTITY_CATEGORY_UNIT | Military or organizational unit. BFO: bfo:ObjectAggregate |
| ENTITY_CATEGORY_PERSON | Individual human being. BFO: bfo:Object |
| ENTITY_CATEGORY_FACILITY | Fixed structure or installation. BFO: bfo:Object |
| ENTITY_CATEGORY_EQUIPMENT | Non-self-propelled system or device. BFO: bfo:Object |
| ENTITY_CATEGORY_MUNITION | Weapon or munition in flight. BFO: bfo:Object |
| ENTITY_CATEGORY_GROUP | Ad-hoc collection treated as single operational entity. BFO: bfo:ObjectAggregate |
| ENTITY_CATEGORY_EVENT | Transient occurrence at a location/time. BFO: bfo:Process |
| ENTITY_CATEGORY_CONTROL_MEASURE | Tactical graphic or control measure. BFO: bfo:Site |
| ENTITY_CATEGORY_INFRASTRUCTURE | Linear or distributed physical infrastructure. BFO: bfo:Object |
| ENTITY_CATEGORY_GEOPOLITICAL | Geopolitical entity or territory. BFO: bfo:Site |
| ENTITY_CATEGORY_NATURAL_FEATURE | Natural geographic feature. BFO: bfo:Site |
| ENTITY_CATEGORY_SIGNAL | Electromagnetic emission or signal. BFO: bfo:Process |
| ENTITY_CATEGORY_ORGANIZATION | Non-military organization or network. BFO: bfo:ObjectAggregate |

EntityStatus

Lifecycle status of an entity in the system - tracks operational state. Entities start in ACTIVE. Use INACTIVE for disabled entities.

| Name | Description |
|---------------------------|---|
| ENTITY_STATUS_UNSPECIFIED | |
| ENTITY_STATUS_ACTIVE | Entity is operational and being tracked. |
| ENTITY_STATUS_INACTIVE | Entity exists but is not actively tracked. |
| ENTITY_STATUS_DELETED | Entity has been removed or is no longer relevant. |

Environment

Operating environment / domain.

| Name | Description |
|-----------------------------|-----------------------|
| ENVIRONMENT_UNSPECIFIED | |
| ENVIRONMENT_SPACE | Exoatmospheric |
| ENVIRONMENT_HIGH_ALTITUDE | Above 50,000 ft |
| ENVIRONMENT_MEDIUM_ALTITUDE | 10,000-50,000 ft |
| ENVIRONMENT_LOW_ALTITUDE | Below 10,000 ft |
| ENVIRONMENT_SURFACE | Ground or sea surface |
| ENVIRONMENT_SUBSURFACE | Underwater |
| ENVIRONMENT_UNDERGROUND | Tunnels, bunkers |

IFFResponse

IFF interrogation response status.

| Name | Description |
|--------------------------|-------------|
| IFF_RESPONSE_UNSPECIFIED | |
| IFF_RESPONSE_CORRECT | |
| IFF_RESPONSE_INCORRECT | |
| IFF_RESPONSE_NO_RESPONSE | |

MovementStatus

Ground entity movement state.

| Name | Description |
|-----------------------------|----------------------------------|
| MOVEMENT_STATUS_UNSPECIFIED | |
| MOVEMENT_STATUS_STATIONARY | Not moving, engine state unknown |
| MOVEMENT_STATUS_MOVING | In motion |
| MOVEMENT_STATUS_HALTED | Temporarily stopped |

NavigationStatus

AIS navigation status.

| Name | Description |
|-------------------------------|-------------|
| NAVIGATION_STATUS_UNSPECIFIED | |

| Name | Description |
|-------------------------------------|----------------------------|
| NAVIGATION_STATUS_UNDERWAY | Using engine |
| NAVIGATION_STATUS_AT_ANCHOR | |
| NAVIGATION_STATUS_MOORED | |
| NAVIGATION_STATUS_NOT_UNDER_COMMAND | Unable to maneuver |
| NAVIGATION_STATUS_RESTRICTED | Restricted maneuverability |
| NAVIGATION_STATUS_AGROUND | Engaged in fishing |
| NAVIGATION_STATUS_FISHING | |

PriorityTier

QoS priority tier for entity replication and routing.

| Name | Description |
|---------------------------|-------------|
| PRIORITY_TIER_UNSPECIFIED | |
| PRIORITY_TIER_LOW | |
| PRIORITY_TIER_MEDIUM | |
| PRIORITY_TIER_HIGH | |
| PRIORITY_TIER_CRITICAL | |

ScanType

Radar/emitter scan pattern type.

| Name | Description |
|------------------------|-------------|
| SCAN_TYPE_UNSPECIFIED | |
| SCAN_TYPE_CIRCULAR | |
| SCAN_TYPE_SECTOR | |
| SCAN_TYPE_CONICAL | |
| SCAN_TYPE_RASTER | |
| SCAN_TYPE_AGILE_BEAM | |
| SCAN_TYPE_NON_SCANNING | |
| SCAN_TYPE_IRREGULAR | |

SidcVersion

Version of the MIL-STD-2525 symbology standard.

| Name | Description |
|--------------------------|-----------------------------------|
| SIDC_VERSION_UNSPECIFIED | Version not specified or unknown. |

| Name | Description |
|---------------------|---|
| SIDC_VERSION_2525C | MIL-STD-2525C (15-character alphanumeric SIDC). |
| SIDC_VERSION_2525D | MIL-STD-2525D (20-character numeric SIDC). |
| SIDC_VERSION_CUSTOM | Custom or non-standard symbology format. |

SignalModulation

Signal modulation type.

| Name | Description |
|------------------------------------|---|
| SIGNAL_MODULATION_UNSPECIFIED | |
| SIGNAL_MODULATION_PULSE | Pulsed signal (has pulse width and PRI). |
| SIGNAL_MODULATION_CW | Continuous wave (no pulse modulation). |
| SIGNAL_MODULATION_CHIRP | Linear frequency modulated (chirp) pulse. |
| SIGNAL_MODULATION_FREQUENCY_HOPPED | Frequency hopping (uses frequency_min_hz / frequency_max_hz range). |
| SIGNAL_MODULATION_PHASE_CODED | Phase-coded pulse (Barker codes, polyphase). |

TerrainType

Terrain classification at entity location.

| Name | Description |
|--------------------------|-------------|
| TERRAIN_TYPE_UNSPECIFIED | |
| TERRAIN_TYPE_URBAN | |
| TERRAIN_TYPE_RURAL | |
| TERRAIN_TYPE_FOREST | |
| TERRAIN_TYPE_DESERT | |
| TERRAIN_TYPE_MOUNTAIN | |

ThreatLevel

Threat severity level (ex: MIDNIGHT).

| Name | Description |
|--------------------------|-------------|
| THREAT_LEVEL_UNSPECIFIED | |
| THREAT_LEVEL_NONE | |
| THREAT_LEVEL_LOW | |
| THREAT_LEVEL_MEDIUM | |
| THREAT_LEVEL_HIGH | |
| THREAT_LEVEL_CRITICAL | |

VerticalMode

Aircraft vertical flight mode.

| Name | Description |
|---------------------------|-------------|
| VERTICAL_MODE_UNSPECIFIED | |
| VERTICAL_MODE_LEVEL | |
| VERTICAL_MODE_CLIMBING | |
| VERTICAL_MODE_DESCENDING | |

VesselType

Vessel type classification for Maritime Domain Awareness (MDA). Aligns with AIS vessel types to enable tracking of commercial traffic that may affect naval operations or indicate anomalous behavior.

| Name | Description |
|--------------------------|-----------------------------------|
| VESSEL_TYPE_UNSPECIFIED | |
| VESSEL_TYPE_CARGO | Merchant cargo vessel |
| VESSEL_TYPE_TANKER | Oil/chemical/gas tanker |
| VESSEL_TYPE_PASSENGER | Cruise, ferry, transport |
| VESSEL_TYPE_FISHING | Commercial fishing vessel |
| VESSEL_TYPE_MILITARY | Naval combatant or auxiliary |
| VESSEL_TYPE_TUG | Tug, tow, or offshore support |
| VESSEL_TYPE_RECREATIONAL | Small craft, yacht (MDA tracking) |
| VESSEL_TYPE_SAR | Search and Rescue |

WakeCategory

ICAO wake turbulence category.

| Name | Description |
|---------------------------|---------------------|
| WAKE_CATEGORY_UNSPECIFIED | |
| WAKE_CATEGORY_LIGHT | < 7,000 kg |
| WAKE_CATEGORY_MEDIUM | 7,000 - 136,000 kg |
| WAKE_CATEGORY_HEAVY | > 136,000 kg |
| WAKE_CATEGORY_SUPER | C-5M Galaxy, An-124 |

Provenance

Provenance

Data lineage and source attribution for trust assessment.

Provenance tracks where information originated, enabling consumers to assess reliability and make informed fusion decisions. Critical for intelligence analysis where source credibility affects confidence.

Loosely modeled on W3C PROV-DM (<https://www.w3.org/TR/prov-dm/>), but intentionally simplified so it can be attached cheaply to every record.

| Field | Type | Description |
|----------------|---------------------------|---|
| id | string | Unique identifier for this provenance record. Enables tracking when same source provides multiple updates. |
| name | string | Human-readable name of the originating source. Examples: “USS Roosevelt CIC”, “SIGINT Station Alpha”, “National Ground Intelligence Center”, “UAV-042 EO/IR” |
| description | string | Additional context about the source or collection method. May include sensor mode, collection geometry, or analyst notes. |
| updated_at | google.protobuf.Timestamp | Timestamp when this information was produced or last updated at the source. Distinct from message transmission time. |
| request_id | string | Server-generated UUID for this specific publish/update request. Used for deterministic LWW (Last-Writer-Wins) conflict resolution when updated_at timestamps are equal. This field is read-only and automatically populated by the server on each request. |
| source_node_id | string | UUID of the RDP (Raft Data Platform) node where this request originated. Used for tracking data flow through the distributed system and assists in conflict resolution and debugging. This field is read-only and automatically populated by the server. |
| producer | string | Identifier of the component that emitted or last transformed this record. Distinct from name, which identifies the operational source (a person, unit, or sensor); <code>producer</code> identifies the technical thing, e.g., a service, algorithm, model, plugin, agent, or client application, that handled the record immediately before publication. Convention: be as precise as possible. Examples: - “my-tracker-v1.2” - “some-fusion-algorithm-v3” - “example-cot-plugin-0.4” - “yolo-v8-object-detection” - “acme-c2-client-2026.1” For records created entirely by hand (e.g., an operator typing in a client), the producer should be client application itself (e.g., the user-agent), and not unset. This is NOT the place for the operational source identity; use <code>id</code> and <code>name</code> for that. |

Reconciliation

ReconciliationMetadata

Reconciliation engine decision metadata.

Stamped on every entity that passes through the reconciliation engine. Tracks which policy made the decision, what action was taken, and why. Read-only; populated by the reconciliation engine on every upsert.

| Field | Type | Description |
|-----------|----------------------|---|
| policy_id | string | Identifier of the reconciliation policy that produced this entity state. Examples: “lww”, “priority_rules”, “custom:/path/to/policy”. |
| action | ReconciliationAction | The reconciliation action that was applied. For entities visible after reconciliation, this is always ACCEPT. |

| Field | Type | Description |
|--------|--------|---|
| reason | string | Human-readable explanation of the reconciliation decision. Examples: “newer timestamp”, “SIGINT (priority 300) > COP_FUSED (priority 100)”. |

ReconciliationAction

Action taken by the reconciliation engine on an entity upsert.

| Name | Description |
|-----------------------------------|---|
| RECONCILIATION_ACTION_UNSPECIFIED | |
| RECONCILIATION_ACTION_ACCEPT | Incoming entity accepted and written. |
| RECONCILIATION_ACTION_REJECT | Incoming entity rejected; existing entity kept. |

Security

SecurityComponents

SecurityComponents contains the structured classification components

| Field | Type | Description |
|-----------------------|-----------------|-------------|
| classification | string | |
| ownerProducer | repeated string | |
| disseminationControls | repeated string | |
| releasableTo | repeated string | |
| sciControls | repeated string | |
| sapControls | repeated string | |
| nonIcMarkings | repeated string | |
| programNicknames | repeated string | |
| fgiSourceOpen | repeated string | |
| fgiSourceProtected | repeated string | |

SecurityMarking

SecurityMarking represents normalized classification marking Compatible with df-classification ISM format

| Field | Type | Description |
|------------|--------------------|-------------|
| raw | string | |
| components | SecurityComponents | |

Task

Principal

Reference to a person, unit, or system that can request, authorize, or execute tasks.

| Field | Type | Description |
|-------|---------------|-------------------------------------|
| id | string | Unique identifier of the principal. |
| type | PrincipalType | Type of principal. |

Task

A directed action to be performed by a principal (e.g., person, unit, or system).

Tasks represent the command and control (C2) layer: orders, requests, and assignments that drive operations. They connect requestors (who wants something done) with executors (who will do it).

IMPORTANT: Tasks are immutable after creation except for state transitions. If task details need to change, cancel the existing task (set state to TASK_STATE_CANCELLED) and create a new one with the correct information.

Examples: ISR collection request, fire mission, movement order, logistics request, sensor tasking.

| Field | Type | Description |
|-----------------|------------------------|--|
| id | string | Unique identifier (UUID). |
| security | SecurityMarking | Classification and handling caveats. |
| name | string | Short name for the task. |
| description | string | Detailed description of what is to be done. |
| type | string | Task type identifier. Examples: "isr_collection", "fire_mission", "movement", "resupply". |
| state | TaskState | Current state in the task lifecycle. |
| priority | TaskPriority | Execution priority. |
| details | google.protobuf.Struct | Task-specific parameters as structured data. Use this field when task-specific information isn't represented in the base Task fields. Clients can include domain-specific parameters based on their task type (e.g., fire mission parameters, ISR sensor configurations, route waypoints). TYPE IDENTIFIER CONVENTION: For structured data, include an "@type" field to identify the schema type. This helps clients determine which protobuf message to unmarshal the data into. Type URL formats: - WDM extensions: "raft.wdm.ngc2.v1.ext.{MessageName}" See raft/wdm/ngc2/v1/ext for available extension types. - Custom types: Use reverse-DNS format (e.g., "com.acme.CustomTaskData") - No @type: Generic/unstructured data where clients inspect fields directly Examples: WDM extension (Fire mission): { "@type": "raft.wdm.ngc2.v1.ext.Fire", "fire": { "target_location": {...}, "rounds": 6, "shell_type": "HE" } } Custom type (ISR): { "@type": "com.acme.ISRTaskData", "isr": { "target_id": "...", "sensor_type": "EO", "duration_min": 30 } } No type identifier: { "target_id": "...", "sensor_type": "EO" } |
| assigned_to | Principal | Principal responsible for executing this task. |

| Field | Type | Description |
|---------------|--------------|---|
| requested_by | Principal | Principal that originated the task request. |
| authorized_by | Principal | Principal that approved the task for execution. May differ from requestor (e.g., fires require commander approval). |
| provenance | Provenance | Data lineage and source attribution. |
| progress | TaskProgress | Current execution progress. Updated via UpdateTaskState. Only meaningful in active states (IN_PROGRESS, PAUSED). |

TaskProgress

Execution progress within a task's current state.

| Field | Type | Description |
|----------------|---------------------------|--|
| phase | string | Execution phase within the current state. Free-form string defined by the client. The server treats this value as opaque. Convention: UPPER_SNAKE_CASE (e.g., "SHOT", "SPLASH", "ON_STATION"). |
| status_message | string | Human-readable status message describing current progress. |
| metadata | google.protobuf.Struct | Additional structured progress data. Follows the same convention as Task.details. |
| updated_at | google.protobuf.Timestamp | Server-assigned timestamp of when this progress was recorded. Read-only; any client-provided value is overwritten by the server. |

PrincipalType

Classification of principal types.

| Name | Description |
|----------------------------|---|
| PRINCIPAL_TYPE_UNSPECIFIED | |
| PRINCIPAL_TYPE_ENTITY | Reference to an Entity in the WDM (unit, platform, person). |
| PRINCIPAL_TYPE_SYSTEM | External system or service. |
| PRINCIPAL_TYPE_USER | Human user account. |

TaskPriority

Task execution priority.

Aligns with military message precedence levels. Higher priority tasks should preempt lower priority work.

| Name | Description |
|---------------------------|--|
| TASK_PRIORITY_UNSPECIFIED | |
| TASK_PRIORITY_LOW | Routine - normal operations, no time constraint. |

| Name | Description |
|------------------------|--|
| TASK_PRIORITY_MEDIUM | Priority - important, expedite handling. |
| TASK_PRIORITY_HIGH | Immediate - urgent, immediate action required. |
| TASK_PRIORITY_CRITICAL | Flash - critical, highest precedence. |

TaskState

Task lifecycle states.

State transitions follow a general flow: DRAFT -> PENDING -> APPROVED -> ASSIGNED -> ACKNOWLEDGED -> PLANNED -> IN_PROGRESS -> COMPLETED

Terminal states (final, cannot be changed): - COMPLETED: Successfully finished - FAILED: Execution failed - CANCELLED: Cancelled/deleted (the deletion mechanism for tasks) - REJECTED: Refused by assignee

| Name | Description |
|-------------------------|---|
| TASK_STATE_UNSPECIFIED | |
| TASK_STATE_DRAFT | Being composed - not yet submitted. |
| TASK_STATE_PENDING | Submitted, awaiting approval. |
| TASK_STATE_APPROVED | Approved by authority, not yet assigned. |
| TASK_STATE_ASSIGNED | Assigned to executing principal. |
| TASK_STATE_ACKNOWLEDGED | Executing principal acknowledged receipt of task. |
| TASK_STATE_PLANNED | Executing principal has developed execution plan. |
| TASK_STATE_IN_PROGRESS | Currently being executed. |
| TASK_STATE_PAUSED | Execution temporarily halted. |
| TASK_STATE_COMPLETED | Successfully completed (TERMINAL STATE). |
| TASK_STATE_FAILED | Failed to complete (TERMINAL STATE). |
| TASK_STATE_CANCELLED | Cancelled/deleted (TERMINAL STATE). This is the deletion mechanism for tasks - there is no separate delete operation. |
| TASK_STATE_REJECTED | Rejected by assignee - unable/unwilling to execute (TERMINAL STATE). |

Scalar Value Types

| .proto Type | Notes | C++ | Java | Python | Go | C# | PHP | Ruby |
|-------------|-------|--------|--------|--------|---------|--------|-------|-------|
| double | | double | double | float | float64 | double | float | Float |
| float | | float | float | float | float32 | float | float | Float |

| .proto Type | Notes | C++ | Java | Python | Go | C# | PHP | Ruby |
|-------------|---|--------|------|----------|--------|-------|----------------|--------------------------------|
| int32 | Uses variable-length encoding. Inefficient for encoding negative numbers – if your field is likely to have negative values, use sint32 instead. | int32 | int | int | int32 | int | integer | Bignum or Fixnum (as required) |
| int64 | Uses variable-length encoding. Inefficient for encoding negative numbers – if your field is likely to have negative values, use sint64 instead. | int64 | long | int/long | int64 | long | integer/string | Bignum |
| uint32 | Uses variable-length encoding. | uint32 | int | int/long | uint32 | uint | integer | Bignum or Fixnum (as required) |
| uint64 | Uses variable-length encoding. | uint64 | long | int/long | uint64 | ulong | integer/string | Bignum or Fixnum (as required) |
| sint32 | Uses variable-length encoding. Signed int value. These more efficiently encode negative numbers than regular int32s. | int32 | int | int | int32 | int | integer | Bignum or Fixnum (as required) |
| sint64 | Uses variable-length encoding. Signed int value. These more efficiently encode negative numbers than regular int64s. | int64 | long | int/long | int64 | long | integer/string | Bignum |

| .proto Type | Notes | C++ | Java | Python | Go | C# | PHP | Ruby |
|-------------|--|--------|------------|-------------|--------|------------|----------------|--------------------------------|
| fixed32 | Always four bytes. More efficient than uint32 if values are often greater than 2^{28} . | uint32 | int | int | uint32 | uint | integer | Bignum or Fixnum (as required) |
| fixed64 | Always eight bytes. More efficient than uint64 if values are often greater than 2^{56} . | uint64 | long | int/long | uint64 | ulong | integer/string | Bignum |
| sfixed32 | Always four bytes. | int32 | int | int | int32 | int | integer | Bignum or Fixnum (as required) |
| sfixed64 | Always eight bytes. | int64 | long | int/long | int64 | long | integer/string | Bignum |
| bool | | bool | boolean | boolean | bool | bool | boolean | TrueClass/FalseClass |
| string | A string must always contain UTF-8 encoded or 7-bit ASCII text. | string | String | str/unicode | string | string | string | String (UTF-8) |
| bytes | May contain any arbitrary sequence of bytes. | string | ByteString | str | []byte | ByteString | string | String (ASCII-8BIT) |