Foundational Architecture

Managed documentation for Foundational Architecture.

- Architecture Overview
- OpenBus
- OpenKnowledge and Semantic Layer

Architecture Overview

Architecture Overview

Field	Value
Distribution	BSS — OpenSOF
Product	OpenSOF
Release	0.15.1
Deployment	`opensof.bss.dev`
Source	`/srv/bss/releases/OpenSOF-v0.15.0-20app`
Evidence	Static release scan
Source fingerprint	`cda583b42f2a0d296b268ab5b174932a378c8be6fb52ffa1af14af896e990f6e`
Status	Generated baseline — human review required

> **Verification boundary:** This page combines platform design guidance with static evidence from the release. It does not prove that every detected interface is enabled, reachable, secure, or operational in the deployed environment.

Layered model

Layer	Representative capabilities	Responsibility
Experience	OpenCOP, OpenEyes, OpenTask, OpenChat, OpenIIA	Operator views, collaboration, investigation and tasking.
Domain services	OpenTrack, OpenRF, OpenSSA, OpenLVC, OpenAnalytics	Mission functions, sensor exploitation, simulation and analysis.
Knowledge & automation	OpenKnowledge, OpenLLM, OpenDAA	Semantic types, queries, models, pipelines and decisions.
Exchange & data	OpenBus, OpenData, OpenFiles, OpenMail	Events, APIs, storage, artifacts and governed dissemination.

Layer	Representative capabilities	Responsibility
Edge & infrastructure	OpenPNT, OpenNetworks, sensors, SDRs, robots, UAS	Time, position, transport, compute and physical interfaces.

Architectural principles

- Modular deployment: components can be deployed, replaced and integrated independently.
- Explicit contracts: APIs, messages, semantic types, time and provenance should be defined rather than implied.
- Edge-first exploitation: process raw streams near the source when practical, then exchange compact operational products.
- Human-machine teaming: automation proposes, correlates and acts through governed workflows with visible status and audit.
- Open integration: customers can combine platform modules with existing C5ISR, analytic, collaboration and enterprise systems.

Observed implementation indicators

Indicator	Observed
Languages	PHP, JavaScript, Python, C, Shell, C++
Frameworks	—
Container definitions	0
Service definitions	9
Nginx configurations	0

Maintainer Notes

Add human-reviewed deployment notes, corrections, decisions, screenshots, and links here. Content outside the generated block is preserved on future runs.

OpenBus

OpenBus

Field	Value
Distribution	BSS — OpenSOF
Product	OpenSOF
Release	0.15.1
Deployment	`opensof.bss.dev`
Source	`/srv/bss/releases/OpenSOF-v0.15.0-20app`
Evidence	Catalog architecture plus release evidence
Source fingerprint	`cda583b42f2a0d296b268ab5b174932a378c8be6fb52ffa1af14af896e990f6e`
Status	Generated baseline — human review required

> **Verification boundary:** This page combines platform design guidance with static evidence from the release. It does not prove that every detected interface is enabled, reachable, secure, or operational in the deployed environment.

Role

OpenBus is the event and message fabric that decouples producers from consumers. A sensor, application or analytic publishes an operational event once; authorized subscribers can consume, enrich, correlate, persist or act on it without forcing a direct integration with every producer.

Recommended event envelope

Field	Purpose

id	Globally unique event identifier.
type	Semantic event/entity type, preferably linked to OpenKnowledge.
source	Producer identity and originating sensor/application.
time	Observation, creation and receipt times with clock-quality metadata.
correlation	Mission, track, task, case, release group or workflow context.
provenance	Processing lineage, transformations and custody.
payload	Domain data or reference to a larger product in OpenFiles/OpenData.
policy	Handling, releasability, retention and access controls.

Integration rule

Use OpenBus for asynchronous operational state and workflow events; use REST for request/response management operations; use WebSockets for low-latency browser or device sessions; use OpenFiles for larger artifacts. Avoid turning the bus into an untyped dump of arbitrary JSON.

Release evidence

Status	Score	Evidence items
detected	105	12

Evidence type	Source	Match
path	`OPENBUS.md`	`openbus`
path	`drivers/openbus_agent.py`	`openbus`
path	`edge/openlvc/rti_openbus_gateway.py`	`openbus`
path	`examples/opencybersec/openbus-security-event.json`	`openbus`
path	`lib/open-networks/openbus.js`	`openbus`
content	`API.md`	`openbus`
content	`CHANGELOG.md`	`openbus`
content	`DEPLOY-GODADDY.md`	`openbus`

Evidence type	Source	Match
content	`MERGE-v0.13.1.md`	`openbus`
content	`OPENBUS.md`	`openbus`
path	`open-bus/app.js`	`open-bus`
path	`open-bus/index.html`	`open-bus`

Maintainer Notes

Add human-reviewed deployment notes, corrections, decisions, screenshots, and links here. Content outside the generated block is preserved on future runs.

OpenKnowledge and Semantic Layer

OpenKnowledge and Semantic Layer

Field	Value
Distribution	BSS — OpenSOF
Product	OpenSOF
Release	0.15.1
Deployment	`opensof.bss.dev`
Source	`/srv/bss/releases/OpenSOF-v0.15.0-20app`
Evidence	Catalog architecture and semantic file scan
Source fingerprint	`cda583b42f2a0d296b268ab5b174932a378c8be6fb52ffa1af14af896e990f6e`
Status	Generated baseline — human review required

> **Verification boundary:** This page combines platform design guidance with static evidence from the release. It does not prove that every detected interface is enabled, reachable, secure, or operational in the deployed environment.

Role

OpenKnowledge provides the shared semantic representation for entities, relationships, observations, tasks, capabilities, effects and evidence. RDF/OWL/SHACL assets and SPARQL endpoints can make that representation queryable and governable across applications.

Why semantics belong in the core

Without a shared meaning layer, integration stops at syntax: two systems may exchange JSON yet disagree on what an entity, position, confidence, task or observation means. Semantic identifiers and constraints let automation select valid actions, let analytics combine compatible data, and give AI a bounded operational vocabulary.

Observed semantic evidence

Measure	Observed
Semantic files	22

Files	`.opennetworks-backup-20260827T210646Z/ontology/openaar-mr.ttl`
	`.opennetworks-backup-20260827T210646Z/ontology/opencybersec.ttl`
	`.opennetworks-backup-20260827T210646Z/ontology/opendaa-openanalytics.ttl`
	`.opennetworks-backup-20260827T210646Z/ontology/opendaa-opendata.ttl`
	`.opennetworks-backup-20260827T210646Z/ontology/openiia.ttl`
	`.opennetworks-backup-20260827T210646Z/ontology/opensof-applications.ttl`
	`.opennetworks-backup-20260827T210646Z/ontology/opensof-core.ttl`
	`.opennetworks-backup-20260827T210646Z/ontology/opensof-ontology.ttl`
	`ontology/openaar-mr.ttl`
	`ontology/opencybersec.ttl`
	`ontology/opendaa-openanalytics.ttl`
	`ontology/opendaa-opendata.ttl`
	`ontology/openiia.ttl`
	`ontology/opennetworks.ttl`
	`ontology/opensof-applications.ttl`
	`ontology/opensof-core.ttl`
	`ontology/opensof-ontology.ttl`
	`third_party/ontowiki/application/config/SysBase.rdf`
	`third_party/ontowiki/extensions/community/insert.sparql`
	`third_party/ontowiki/extensions/datagathering/SyncSchema.rdf`
	`third_party/ontowiki/extensions/exconf/resources/Examples.rdf`
	`third_party/ontowiki/extensions/exconf/resources/PluginRepository.rdf`

Term indicators	RDF: 4239, OWL: 2287, ontology: 2031, provenance: 271, SPARQL: 258, SHACL: 143, JSON-LD: 18

Governance checklist

- Assign stable URIs to core entity, relationship, observation, task and effect types.
- Version ontologies and record compatibility/migration rules.
- Validate critical payloads with SHACL or equivalent constraints.
- Record provenance and source confidence separately from asserted facts.
- Protect write access to production knowledge while allowing broad governed query access.

Maintainer Notes

Add human-reviewed deployment notes, corrections, decisions, screenshots, and links here. Content outside the generated block is preserved on future runs.