SURF

A Year of Workflow Orchestrator

June 2025 → June 2026

Hans Trompert (SURF)

June 8, 2026, at TNC26

The year in numbers (human work, bots excluded)

- 1,880 human commits on the combined set of repositories
  - plus 735 automated dependency commits (renovate/dependabot)
- 663 human-authored merged PRs
- 40 actively developed repositories, of which 9 brand-new this year
- Notable major releases:
  - orchestrator-core 4 → 5
  - orchestrator-ui-library 5 → 8
  - 70+ releases each (we believe in continues integration)

The ecosystem at a glance

Five themes, one platform:

- **Core framework** - the orchestration engine
- **Frontend & forms** - the operator UI
- **AI & agents** - new this year, and everywhere
- **NSI network automation** - a new stack
- **Device & network tooling** - the integration edge

...on a foundation of shared libraries and tools.

Core framework 4.0 → 5.0

- 354 human commits · 8 feature releases (4.1 → 4.8) leading up to 5.0
  - upgrade guide and script
- Scheduler CRUD API + UI
  - Moved from schedule decorator to database backed APScheduler
  - Schedule types: once, interval, cron
  - Uses core form through `/api/forms` (user extendible)
  - Allows scheduling of workflows **with user inputs**
- Authorization for workflows and input steps
  - @workflow → authorize_callback (start) and retry_auth_callback (resume/retry).
  - @inputstep → resume_auth_callback (resume) and retry_auth_callback (retry)
  - Richer AuthContext, who (user), what they're attempting (action), on which workflow (workflow), and at which step (step) - instead of just who

SURF

Core framework 4.0 → 5.0

- New workflow type `reconcile`
  - push the orchestrator's authoritative state to external systems
- Introduced task gating with `run_predicate`
  - See usage in `no_uncompleted_instance()` used in the core for default tasks
- Search was rebuilt across 4.4–4.8 into AI Search
  - Based on pgvector – vector similarity search for PostgreSQL
  - with five retrievers - structured (filters), fuzzy (trigram), semantic (embeddings), and two hybrids
  - 5.0 made AI Search the default and deprecated the old Postgres TSV search
  - Integrating a LLM for semantic search is optional, default off
- Moved from psycopg2 to psycopg3 (`postgresql+psycopg://`)
  - since psycopg3 auto-begins transactions, they're now centrally owned by `transactional()`
  - and closed before side-effects fire, preventing idle-in-transaction leaks

SURF

Core framework 4.0 → 5.0

- Namespace package (PEP 420)
  - independently-installed add-on modules can live under one orchestrator.* tree
  - Orchestrator-core now uses orchestrator.core
  - Orchestrator-shell is the first package under orchestrator.shell
- Secrets wrapped in Pydantic Secret types
  - DATABASE_URI, CACHE_URI, WEBSOCKET_BROADCASTER_URL, SESSION_SECRET use SecretStr
  - Auto-masked in logs, prints & tracebacks
  - Breaking: read via .get_secret_value()

Frontend & Forms

- activity
  - orchestrator-ui-library: 394 human commits, v5 → v8
  - pydantic-forms-ui: 387 human commits, 0.2 → 2.0
- Frontend renderer swap
  - dropped third-party uniforms (Vazco)
  - adopted in-house pydantic-forms-ui
  - schema hint key `uniforms` → `extraProperties` (breaking change)
- What did not change
  - Forms described with Pydantic models in backend → JSON Schema → frontend renderer
- Advantages of owning the renderer
  - React lib purpose-built for WFO + custom components
  - client-side Zod validation derived from the backend's JSON Schema
  - First additional form field that renders markdown

Frontend & Forms

- New orchestrator-ui-library features
  - RBAC in the GUI – actions are greyed out when not authorized
  - scheduler tab - uses backend-served workflow forms
  - virtualized tables – renders faster and smooth scroll with huge lists
  - step-header/detail overrides on processes page
  - Process notes
  - Editing of metadata descriptions (products, ...)
  - Subscription quick actions
- Soon
  - new structured search bar to compose complex queries

AI

- orchestrator-agent — natural-language search agent for WFO
  - Ask plain-language questions over subscriptions, products, workflows, processes
  - LLM-driven (pydantic-ai/GPT-4o); Planner → 4 skills (Search/Aggregate/Actions/Text)
  - Reuses core's search engine; on zero hits, broadens to a filterless semantic pass
  - One app, three protocols: AG-UI, A2A, MCP
  - Container (ghcr), OIDC-secured
- MCP endpoint now in five services
  - orchestrator-agent – see above
  - orchestrator-core - subscriptions, products, workflows, processes
  - nsi-aggregator-proxy - NSI multi domain circuits and circuit segments
  - SuPA - NSI per domain circuit segments
  - influxdb3_mcp_server (fork) - network service metrics

SURF

NSI network automation

- Use case: ANA-GRAM - automate and orchestrate L2 circuit provisioning over ANA trans-Atlantic links
- nsi-dds-proxy
  - REST view of topology (STPs/SDPs/switching services) from DDS documents
- nsi-aggregator-proxy
  - REST interface to NSI CS v2 SOAP, single state machine, auto-commit, async callbacks
- nsi-orchestrator (wip)
  - WFO-based lifecycle of topologies/switching services/STP/SDP/MDP2P
- nsi-mgmt-info (wip)
  - overviews & statistics for NSI multi domain circuit monitoring and planning
- Authentication/Autorisation
  - nsi-auth - mTLS certificates
  - ana-automation-ui - OIDC portal (wip)

Device & network tooling

- lso
  - Added synchronous /api/execute, playbook progress, fire-and-forget callback
  - Breaking changes
    - Ansible 10→13 (retest playbooks!)
    - API v2.0 replaced HTTP 404 (Not Found) with 410 (Gone) and 503 (Service Unavailable)
- gnmic-cluster-chart
  - Helm chart to deploy autoscaling gNMIc cluster → Kafka
  - Known to work with ~300 routers, ~2B events/day
- orchestrator-optical
  - Products + workflows + migrations for optical gear (Infinera FlexILS / TL1)
  - First candidate add-on module
  - Only needs to add namespace orchestrator.optical packaging

The end.

SURF