OC4IDS Portal Development

Workflow Guide: From Field Mapping to Launch

A step-by-step guide for developing OC4IDS-compliant infrastructure transparency portals that meet the Publication Criteria.

Typical Timeline: **14 to 21 weeks**

Phases: 6 sequential phases across Discovery, Build, and Deliver

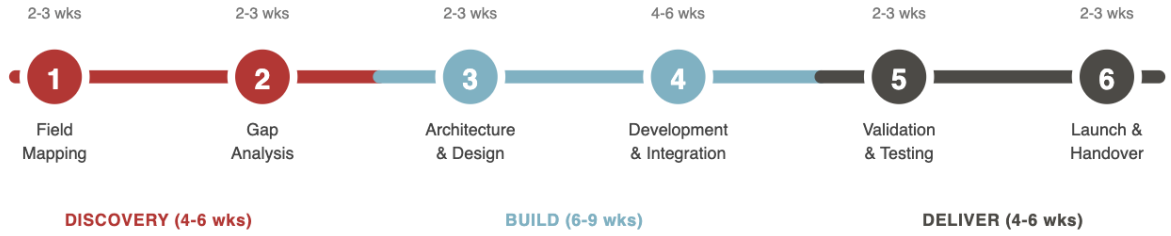
Audience: Country programme managers, technical consultants, government IT teams

CoST – the Infrastructure Transparency Initiative
Open Data Systems

This document outlines the standard workflow for developing an OC4IDS-compliant infrastructure transparency portal that meets the OC4IDS Publication Criteria. The process ensures data quality, schema compliance, and system durability from inception through launch. The Publication Criteria define 11 pass/fail checks covering registration, discoverability, retrievability, validity, conformance, and documentation. Every phase in this workflow maps to one or more of those criteria.

TYPICAL TIMELINE: 14 TO 21 WEEKS

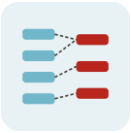
Government approval cycles, data access delays, and stakeholder availability consistently extend timelines beyond minimum estimates. Plan for the upper range.



Pre-requisites

Before beginning Phase 1, confirm all of the following are in place:

✓	Stakeholders identified, including the System Owner (a senior technocrat who survives political turnover)
✓	Use cases workshopped with target user groups (civil society, procuring entities, oversight bodies)
✓	Policy expert identified (familiar with infrastructure project and contract legislation)
✓	Technical expert identified (familiar with government data sources and systems)
✓	OC4IDS technical expert identified (understands OC4IDS data structures, codelists, and publication requirements)
✓	OC4IDS prefix registered with the OC4IDS Helpdesk (required for valid project identifiers under Publication Criterion 1)



PHASE 1 | 2 to 3 weeks

Field-Level Mapping

Align existing government data systems with OC4IDS schema requirements.

KEY ACTIVITIES

1. Document all existing data fields in source systems (procurement, budget, project management, contract management databases)
2. Map source fields to OC4IDS fields using the OC4IDS mapping spreadsheet. Three actors required: a technical expert familiar with data sources, a policy expert familiar with infrastructure/contract legislation, and an OC4IDS technical expert
3. Identify direct mappings, transformations required, and unmappable fields. For unmappable fields, determine whether the gap is data collection (field exists but is empty) or schema (no OC4IDS field exists)
4. Map all relevant codelists: projectStatus, contractingProcessStatus, projectType, contractNature, sector, documentType, partyRole, procurementMethod, and any open codelists where source values need alignment
5. Confirm terminology alignment: title (not project_name), status (not project_status), supplier (not contractor/vendor), tender (not bid/RFP), tenderer (not bidder), publication (not disclosure)
6. Register OC4IDS identifier prefix with the OC4IDS Helpdesk if not already done

DELIVERABLES

✓	Field Mapping Matrix (source system fields to OC4IDS paths with transformation logic)
✓	Codelist Alignment Table (source values to OC4IDS codelist values for all relevant codelists)
✓	Unmappable Fields Register (fields with no OC4IDS equivalent, candidates for extension proposal)

PHASE 2 | 2 to 3 weeks

Gap Analysis & Data Assessment

Identify data gaps between current systems and CoST IDS publication requirements.

KEY ACTIVITIES

1. Audit data completeness against CoST IDS proactive disclosure items (40 items) and reactive disclosure items (27 items) using the CoST IDS to OC4IDS mapping
2. Assess data quality against all six dimensions: Validity, Accuracy, Consistency, Timeliness, Uniqueness, Completeness
3. Measure OC4IDS field coverage (Publication Criteria Metric 7) and CoST IDS element coverage (Metric 8) to establish a baseline
4. Identify missing data points and the collection mechanisms needed to fill gaps (manual entry forms, system integrations, periodic imports)
5. Evaluate extension requirements for country-specific fields. Extensions are a last resort: first verify no existing OC4IDS field accommodates the data
6. Document data governance expectations: who updates which fields, at what frequency, and through which system

DELIVERABLES

✓	Gap Analysis Report (coverage against CoST IDS proactive and reactive items)
✓	Data Quality Assessment (baseline scores across all six dimensions)
✓	CoST IDS Coverage Baseline (field-level and element-level percentages)
✓	Extension Requirements Specification (if needed, with Helpdesk escalation draft)

PHASE 3 | 2 to 3 weeks

System Architecture & Design

Design the technical infrastructure for OC4IDS-compliant publication.

KEY ACTIVITIES

1. Define data flow architecture: source systems to ETL to portal database to API to frontend. Include data sovereignty and hosting decisions
2. Design API endpoints aligned with OC4IDS JSON structure. Include: project package, individual project, bulk download, and pagination for automated retrieval
3. Specify data validation rules to be applied on ingest (schema validation using lib-cove-oc4ids or equivalent)
4. Specify red flag detection logic and automated alert rules (when red flags and automation are part of portal design)
5. Conduct user workshops with target audiences to define frontend visualization requirements and use cases
6. Plan user interface for project publication, search, and filtering functionality based on workshop outputs
7. Formalize governance structure with the identified System Owner. Define roles for data entry, quality assurance, system administration, and publication approval

DELIVERABLES

✓	Technical Architecture Document (including hosting and data sovereignty decisions)
✓	API Specification (with bulk download and automated retrieval endpoints)
✓	Data Validation Rules Specification (shared resource for developers)
✓	Red Flag Detection Rules (if applicable)
✓	UI/UX Wireframes (informed by user workshop findings)
✓	Data Governance Framework (roles, update frequencies, approval workflows)

PHASE 4 | 4 to 6 weeks

Development & Integration

Build the portal with OC4IDS schema compliance embedded from day one.

KEY ACTIVITIES

1. Implement ETL pipelines with OC4IDS transformation logic based on the Field Mapping Matrix from Phase 1
2. Build API layer with JSON schema validation using lib-cove-oc4ids to check validity programmatically on every data ingest
3. Develop frontend with project search, filtering, and visualization capabilities based on Phase 3 wireframes
4. Implement bulk download functionality in both JSON (OC4IDS project packages) and tabular (CSV/spreadsheet) formats

5. Integrate red flag detection and automated alerts (if applicable)
6. Implement data validation on ingest: catch compliance errors at data entry, not publication
7. Run semantic checks against sample data: verify realistic budgets, valid coordinates, correct party role assignments, and standardized sector codes
8. Conduct iterative user testing with government stakeholders during development sprints

DELIVERABLES

✓	Working Portal (development environment)
✓	ETL Pipeline Documentation
✓	API Documentation
✓	Validation Test Results (schema validation, semantic checks)
✓	Bulk Download Functionality (JSON and CSV/spreadsheet formats)

PHASE 5 | 2 to 3 weeks

Validation & Testing

Verify schema compliance, publication criteria, and system functionality before launch.

KEY ACTIVITIES

1. Run full OC4IDS schema validation on all published data: zero validation errors required (Criterion 10) and zero structure warnings required (Criterion 11)
2. Verify the data and portal satisfy all 11 Publication Criteria and Checks
3. Execute all 8 semantic checks: standardized sectors, realistic names, realistic budgets and values, realistic funder names, valid dates, correct role assignments, valid coordinates
4. Verify data is retrievable programmatically (Criterion 3): test the scraper/API automation end to end
5. Conduct user acceptance testing with government stakeholders and target user groups
6. Performance testing (page load times, API response times under load) and security testing
7. Review CoST IDS coverage metrics and document any remaining gaps with justification

DELIVERABLES

✓	OC4IDS Validation Report (zero errors, zero structure warnings)
✓	Publication Criteria Compliance Report (pass/fail on all 11 criteria)
✓	Semantic Checks Report
✓	User Acceptance Sign-off
✓	Security Assessment Report
✓	CoST IDS Coverage Report (final metrics with gap justifications)

PHASE 6 | 2 to 3 weeks

Launch & Handover

Deploy to production and transfer ownership to System Owner.

KEY ACTIVITIES

1. Deploy to production environment with SSL/TLS configured
2. Ensure portal homepage is indexed by web search engines (Criterion 2: discoverability)
3. Publish the Publication Policy and Data User Guide alongside the portal (Criterion 8)
4. Register the portal with the CoST Datastore for ongoing monitoring
5. Train government IT staff on system administration, backup procedures, and incident response
6. Train data entry staff on OC4IDS requirements, terminology, and codelist usage
7. Establish ongoing support and maintenance protocols with defined SLAs
8. Document lessons learned and replication guidance for other country programs
9. Confirm the data has an open data license (e.g., CC-BY-4.0) published alongside the portal

DELIVERABLES

✓	Live Production Portal (passing all Publication Criteria)
✓	Publication Policy and Data User Guide
✓	Training Materials and Completion Records
✓	Operations and Maintenance Manual
✓	Helpdesk Support Protocol
✓	Open Data License Statement

Critical Success Factors

1 **Field mapping BEFORE development.**
Retrofitting OC4IDS fields post-launch is 10x harder and introduces data quality debt that compounds with every publication cycle.

2 **Identify System Owner before Phase 1.**
This must be a senior technocrat (Permanent Secretary, Head of IT) who survives political turnover. The system must outlast any single administration.

3 **Schema validation on ingest.**
Catch compliance errors at data entry, not publication. A portal that publishes invalid data damages credibility faster than one that publishes nothing.

4 **Use OC4IDS terminology from day one.**
title, status, supplier, tender (not project_name, contractor, bid). Terminology alignment prevents mapping errors and training confusion.

5 **Map all codelists, not just status fields.**
projectType, contractNature, sector, documentType, and partyRole are equally important for interoperability and data quality.

6 **Plan for the full Publication Criteria from inception.**
Discoverability, retrievability, bulk downloads, and documentation are not afterthoughts. They are pass/fail requirements.

Key References

Resource	URL
OC4IDS Schema Reference	https://standard.open-contracting.org/infrastructure/latest/en/reference/
Publication Criteria	https://docs.google.com/document/d/14o1TqWDu46bicNFC6olGBYU22gyl9QbCXdDxXBqNQFE
CoST IDS to OC4IDS Mapping	https://standard.open-contracting.org/infrastructure/latest/en/cost/
lib-cove-oc4ids (Validation)	https://github.com/open-contracting/lib-cove-oc4ids
OC4IDS Data Review Tool	https://review-oc4ids.standard.open-contracting.org/
Prefix Registration	https://standard.open-contracting.org/infrastructure/latest/en/reference/prefixes/
OC4IDS Codelists	https://standard.open-contracting.org/infrastructure/latest/en/reference/codelists/