Software Requirements Specification

for

Federation Architecture for Composed Infrastructure Services (FACIS)

Multi-Party Service Level Agreement Demonstrator

FACIS.SLA.Demo

Version 1.0 (November 2026)

Published by
eco – Association of the Internet Industry (eco – Verband der Internetwirtschaft e.V.)
Lichtstrasse 43h
50825 Cologne, Germany
Copyright © eco Association of the Internet Industry (eco – Verband der Internetwirtschaft e.V.)
This work is licensed under the Creative Commons Attribution 4.0 International License.
To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/ or send a letter to
Creative Commons, PO Box 1866, Mountain View, CA 94042, USA



© 2026. This work is licensed under a CC BY 4.0 license.

Table of Contents

1 Introduction 1
1.1 Purpose 1
1.2 Background: FACIS SLA Taxonomy and Playbook 1
1.3 Product scope 1
1.4 Definitions and abbreviations 2
1.5 References 3
1.6 Conformance language 3
2 Overall Description 4
2.1 Coordination model: lead provider with SLA Master 4
2.2 Component stack 4
2.3 Participants and roles 7
2.4 Contract structure 7
2.5 Process overview 8
2.6 Constraints 8
2.7 Dependencies 9
2.8 User documentation 9
3 Functional Requirements 10
3.1 Phase I – Selection (discovery) 10
3.2 SLA templates 14
3.3 Phase II – Contract initiation 15
3.4 Phase III – Contract negotiation (simplified) 18
3.5 Phase IV – Signing 21
3.6 Storage, export and audit 23
3.7 Identity, credentials and entitlement 25
3.8 Orchestration and extensibility 26
4 Non-Functional Requirements 28
5 Demonstration Scenario 28

© 2026. This work is licensed under a CC BY 4.0 license.

1 Introduction

1.1 Purpose

This document specifies the functional and non-functional requirements of the FACIS Multi-Party Service Level Agreement Demonstrator (SLA demonstrator). The demonstrator illustrates, at Technology Readiness Level (TRL) 4, how an SLA package between a customer and several cloud and edge providers is discovered, assembled, negotiated and signed in the lead-provider (hub-and-spoke) coordination model of the FACIS SLA Governance Framework, structured according to the FACIS SLA Taxonomy and aligned with the Playbook.

Of the three coordination models of the FACIS SLA Governance Framework (Section 2.1), the demonstrator implements only the lead-provider model, because it matches current contracting practice and can be realised with the standard functions of the FACIS Digital Contracting Services (DCS) core. The demonstrator collects feedback from providers and customers as input for the further design. The other two models are out of scope.

The customer contracts only with Provider P1, which bundles the contributions of all further providers in one SLA Master. The screen samples in Section 3 illustrate the required functional content of the user interface; their layout, styling and sample values are not binding.

1.2 Background: FACIS SLA Taxonomy and Playbook

The FACIS SLA Taxonomy describes an SLA through four building blocks:

- **Foundation** – scope, terminology, parties, covered services and their dependencies.
- **Service Level Objectives (SLOs)** – the measurable quality commitments made about a service, each with its assigned Service Level Indicators (SLIs) and corresponding metrics.
- **Monitoring** – how compliance with the SLOs is measured and reported through transparent and well-defined metrics.
- **Enforcement** – remedies, escalation paths and rights in case of non-adherence.

Quality commitments follow the hierarchy SLO → SLI → metric (example: SLO high availability; SLI uptime percentage; metric monthly average uptime ≥ 99.9 %, excluding approved maintenance windows). The Playbook [FACIS-PB] gives practical guidance and template structures for multi-provider SLAs. The Taxonomy and the Playbook determine the structure of the SLA templates; the JSON template of the Taxonomy [FACIS-TAX] is the reference for the SLA content. All SLO categories of the Taxonomy are permitted; the examples in this document use availability and performance.

1.3 Product scope

1.3.1 In scope

- Publication of cloud service descriptions and of provider-specific SLA sub-templates in federated catalogues, and their discovery through FAP Decentralised Catalogue Management (Phase I).
- Creation of the SLA Master template by P1, covering P1's own service commitments and the overall commitments towards C, import of the selected SLA sub-templates into the DCS operated by P1, and assembly of a contract package consisting of an SLA Master and one sub-SLA per provider (Phase II).
- Assignment of SLA elements (SLOs, SLIs, metrics) to the responsible provider; each party accesses only the contracts to which it is a party (Phase II).
- Simplified negotiation: a party of a contract proposes a change to a text element and the other party confirms or modifies it (Phase III).
- Asynchronous, legally binding signing by signers holding a Power of Attorney (PoA) credential; finalisation, archiving and activation of the contract package (Phase IV).
- Identity, credential and entitlement management for all participants with FAP PCI, OCM W-Stack and PCM Cloud.
- Machine-readable SLA content (JSON-LD, ODRL, SHACL) aligned with the FACIS SLA Taxonomy, synchronised with a human-readable PDF/A-3 rendering.

© 2026. This work is licensed under a CC BY 4.0 license.

1.3.2 Out of scope

- Change requests to the FACIS DCS core. The DCS core provides no SLA-specific functions; the demonstrator adds them through configuration, SLA templates, the SLA vocabulary (SHACL shapes in the Semantic Hub), the other XFSC/FACIS components listed in Section 2.2, XFSC ORCE flows and organisational procedures. The DCS functions the demonstrator relies on are listed in Section 2.7
- Software-supported harmonisation of SLOs, SLIs and metrics between providers and automated consistency checks between the SLA Master and the sub-SLAs. In the demonstrator, harmonisation is done manually by P1 with the providers.
- Moderation of sub-declarations owned by individual parties; only the agreed text of each contract is signed.
- Per-signer deadlines and reminders, an explicit decline action for signers, automatic revocation of a signed contract when a signer's credential is revoked, and termination of individual sub-SLAs while the SLA Master continues (partial termination of the package).
- SLA monitoring, measurement ingestion, performance tracking, claim handling and enforcement.
- Automated legal or regulatory compliance validation of SLA content.
- Operation of one DCS instance per participant with DCS-to-DCS exchange; production-grade operation, scalability and availability.
- SLA change management after signing: amendments, change notices and per-provider validity periods.
- The SLA broker and decentralised multi-party coordination models, customer-led and ecosystem-driven initiation, and multi-party template composition..

1.4 Definitions and abbreviations

Term	Definition
Baseline FC	Instance of the XFSC Federated Catalogue operated by a provider to manage its SLA sub-templates.
C	Customer C, receiving party of the SLA; signs the SLA Master.
CAT	XFSC Federated Catalogue for cloud service descriptions.
DCS	Digital Contracting Services: open-source software for automated and legally compliant contract services.
FAP DCM	FAP Decentralised Catalogue Management: harvests, maps and searches several catalogues.
FAP PCI	FAP Principal Credential Issuance: issues PoA credentials.
Lead-provider model	Coordination model of the FACIS SLA Governance Framework (hub-and-spoke): one provider contracts with the customer and with each further provider and coordinates them. The other models are the dedicated SLA broker and decentralised multi-party SLA management.
OCM W-Stack	XFSC Organisational Credential Manager: issues and verifies entitlement credentials.
OID4VCI / OID4VP	OpenID for Verifiable Credential Issuance / for Verifiable Presentations.
ORCE	XFSC orchestration engine; used for workflow webhooks, UIs and issuance flows.
P1 (SPOC)	Provider 1, lead provider and single point of contact for the customer. Operates the DCS and acts as contract creator, reviewer, approver and signer. Shown as CSP (cloud service provider) in figures and screen samples; this document uses P1.
P2, P3	Providers delivering service components. Author SLA sub-templates and sign their sub-SLA with P1. P4–Pn may join following the same pattern.
PCM Cloud	XFSC Personal Credential Manager (cloud wallet) of each user.

© 2026. This work is licensed under a CC BY 4.0 license.

Term	Definition
PoA	Power of Attorney credential that authorises a natural person to sign on behalf of an organisation.
SLA (sub-)template	Machine-readable template in the DCS template format (JSON-LD with ODRL policy). The SLA Master template (DCS template level FRAME_AGREEMENT) is the main element; each provider's sub-template (level SUB_AGREEMENT) is a subordinate element that contains the provider's SLOs with their SLIs and metrics.
SLA Master	Contract between C and P1. Contains P1's own service commitments and the overall commitments towards C (end-to-end SLOs for the covered services), names the provider responsible for each SLO and lists the attached sub-SLAs.
SLO, SLI, metric	Service Level Objective, Service Level Indicator and the measurement rule of an SLI (FACIS SLA Taxonomy).
Sub-SLA	Contract between P1 and one provider (P1↔P2, P1↔P3), linked to the SLA Master as child contract.
TRL	Technology Readiness Level.

1.5 References

ID	Document / repository
[FACIS-TAX]	FACIS – A Taxonomy for SLAs in Federated Service Ecosystems: https://www.facis.eu/wp-content/uploads/2025/11/FACIS-A-Taxonomy-for-SLAs-in-Federated-Service-Ecosystems.pdf
[FACIS-PB]	FACIS SLA Governance Framework Playbook: https://www.facis.eu/wp-content/uploads/2025/11/FACIS-SLA-Governance-Framework-Playbook.pdf
[FACIS.DCS.SRS]	Software Requirements Specification for FACIS.DCS; code: https://github.com/eclipse-xfsc/facis-dcs
[OCM]	XFSC OCM W-Stack: several submodules listed at https://github.com/eclipse-xfsc/org.eclipse.xfsc#xfsc-navigation
[PCM]	XFSC PCM Cloud: several submodules listed at https://github.com/eclipse-xfsc/org.eclipse.xfsc#xfsc-navigation
[CAT]	XFSC Federated Catalogue: https://github.com/eclipse-xfsc/federated-catalogue
[FAP-DCM]	FACIS FAP Decentralised Catalogue Management: https://github.com/eclipse-xfsc/facis/tree/main/FAP/Decentralised%20Catalogue%20Management
[FAP-PCI]	FACIS FAP Principal Credential Issuance: https://github.com/eclipse-xfsc/facis/tree/main/FAP/Principal%20Credential%20Issuance
[RFC 2119]	Key words for use in RFCs to indicate requirement levels

1.6 Conformance language

The key words MUST, MUST NOT, SHOULD, SHOULD NOT and MAY are used as defined in [RFC 2119]. Requirement IDs follow the format [SLA-DEMO-<Area>-<No>] with the areas CAT (catalogue and discovery), TR (templates), CWE (contract initiation and negotiation), SM (signing), CSA (storage and archive), PACM (audit), IAM (identity, credentials and entitlement), ORC (orchestration) and NFR (non-functional requirements). Within this document, requirements are cited in short form (e.g. TR-01 for [SLA-DEMO-TR-01]); [DCS-…] identifiers refer to [FACIS.DCS.SRS]; D-01 to D-07 refer to the dependencies in Section 2.7.

© 2026. This work is licensed under a CC BY 4.0 license.

2 Overall Description

2.1 Coordination model: lead provider with SLA Master

The FACIS SLA Governance Framework describes three coordination models for SLAs in federated service ecosystems: lead provider (hub-and-spoke), dedicated SLA broker and decentralised multi-party SLA management. The demonstrator implements the lead-provider model, which the Taxonomy and the Playbook describe as today's prevailing practice. The customer communicates only with P1 as single point of contact (SPOC). P1 concludes the SLA Master with C and one sub-SLA with each further provider; P2 and P3 coordinate only with P1. Further providers (P4–Pn) join following the same pattern.

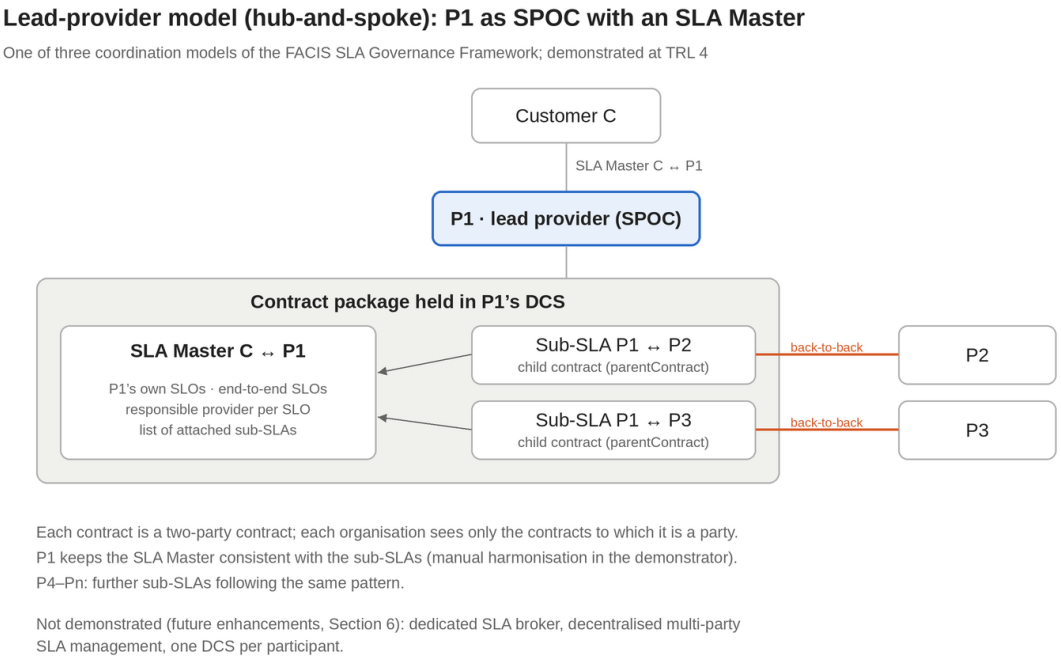


Figure 1: Lead-provider model – P1 as SPOC with an SLA Master and attached sub-SLAs

Contract governance is centralised at P1: P1's DCS holds all contracts of the package. Only the catalogues are decentralised: each provider publishes its service descriptions in CAT and its SLA sub-templates in its own baseline catalogue, and FAP DCM harvests both.

2.2 Component stack

The demonstrator uses the following XFSC and FACIS components as delivered and adds configuration, schemas (SLA vocabulary), templates and ORCE flows. The extensions of components other than the DCS core that this specification requires are part of the demonstrator: the SLA template UI for the baseline catalogues (CAT-02), the service editor on CAT (CAT-01), the search and selection UI of FAP DCM (CAT-04) and the flow UI (ORC-03). The DCS core is not changed: if a DCS function proves insufficient (Section 2.7), the requirement is realised by an ORCE flow or an organisational procedure, or its scope is reduced.

Component	Operated by	Role in the demonstrator
(a) FACIS DCS	P1	Full workflow from SLA Master template creation, template import and contract assembly up to signature: template repository with Semantic Hub, contract workflow engine, signature management, contract storage and archive, process audit and compliance. DCS UI with OID4VP login.
(b) Baseline Federated Catalogue	P2, P3 (one each)	Catalogue in which the provider publishes its SLA sub-templates: versioned asset store (JSON-LD / ODRL), FC API. Sub-templates are authored either with the template editor of a provider-operated DCS instance used only for

© 2026. This work is licensed under a CC BY 4.0 license.

Component	Operated by	Role in the demonstrator
		authoring and published via the DCS template catalogue integration, or with the SLA template UI of the demonstrator; in both cases in the DCS template format (CAT-02).
(c) Federated Catalogue CAT	Ecosystem (demo portal)	Service editor (CAT-01) for cloud service descriptions; schema per Trusted Cloud service description (SHACL / JSON-LD); FC API with query and verification. Holds service descriptions only, not SLA content.
(d) FAP DCM	Ecosystem	Catalogue and schema registry, asset crawler and mapper, local catalogue with harmonised metadata and provenance, search and selection UI (ORCE UI Builder).
(e) FAP PCI	Each organisation (tenant)	Issues PoA credentials for contract signers (organisation, role such as Contract Signer, signature, status entry) via an ORCE issuance flow.
(f) OCM W-Stack	Ecosystem	Issues entitlement credentials that map to DCS roles (OID4VCI); verifies presentations (OID4VP); status list service; trust anchors.
(g) PCM Cloud	Each user	Cloud wallet (web UI) that receives, holds and presents PoA and entitlement credentials.
(h) XFSC ORCE	Shared	DCS workflow webhooks and contract target flows; FAP DCM UIs; FAP PCI issuance flows.

© 2026. This work is licensed under a CC BY 4.0 license.

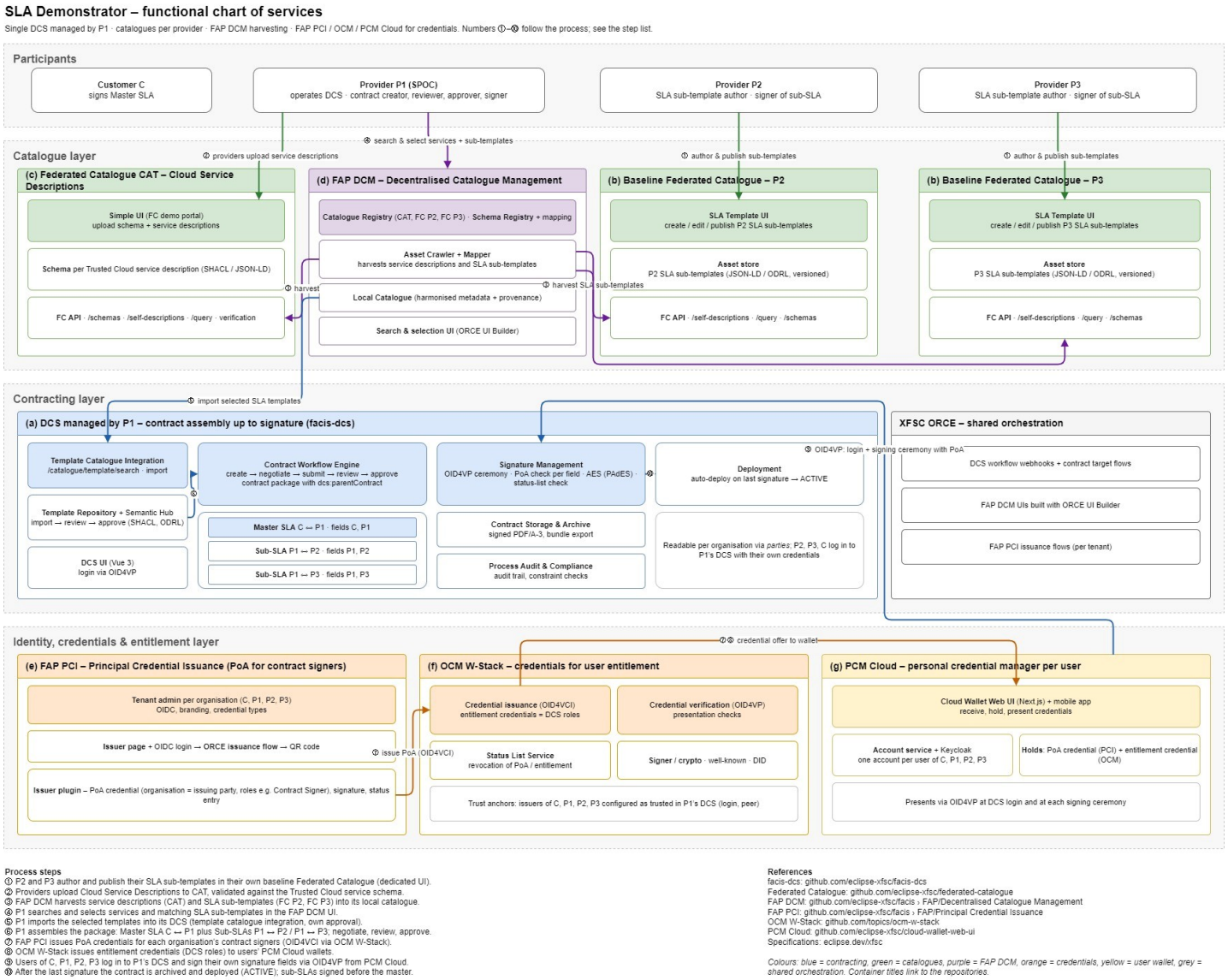


Figure 2: Functional chart of services – participants, catalogue layer, contracting layer and identity layer; numbers 1–10 follow the process in Section 2.5

© 2026. This work is licensed under a CC BY 4.0 license.

2.3 Participants and roles

Participant	Responsibilities	DCS roles (via entitlement credential)	Signs
Customer C	Reviews, negotiates and confirms the SLA Master; communicates only with P1; sees in the SLA Master which providers deliver which SLOs and can reject a sub-provider by a change proposal (CWE-05).	Contract Reviewer / Approver / Signer for the SLA Master	SLA Master
Provider P1 (SPOC)	Operates the DCS; creates the SLA Master template with its own commitments; searches and selects services; imports sub-templates; assembles the package; invites parties; negotiates with C and with the providers; keeps the SLA Master consistent with the sub-SLAs.	Template Manager / Approver, Contract Creator, Contract Manager, Reviewer, Approver, Signer	SLA Master and all sub-SLAs
Provider P2, P3	Publish service descriptions and SLA sub-templates; propose, confirm or modify changes of their own sub-SLA.	Contract Reviewer / Approver / Signer for their sub-SLA	Own sub-SLA
Tenant administrator (per organisation)	Configures FAP PCI for the organisation and issues PoA credentials to signers.	–	–

2.4 Contract structure

The contract package uses the hierarchical contract structures of the DCS core ([DCS-FR-CWE-02], parentContract) and its template levels ([DCS-FR-TR-02]: SLA Master FRAME_AGREEMENT, sub-SLA SUB_AGREEMENT):

- **SLA Master C ↔ P1** – main element: foundation (parties, covered services and, as text, their dependencies, definitions); the SLOs towards C with their SLIs and metrics, including those that P1 delivers itself, and per SLO the responsible provider; monitoring and reporting; enforcement and remedies; the list of attached sub-SLAs. Parties: C, P1.
- **Sub-SLA P1 ↔ P2** – subordinate element: SLOs, SLIs and metrics of P2's service component with the monitoring, reporting and enforcement provisions that apply to them, taken from P2's SLA sub-template. Parties: P1, P2.
- **Sub-SLA P1 ↔ P3** – as above for P3. Further providers add further sub-SLAs.

Each contract is a two-party contract and fits the party model of the DCS core; the package is a bundle of bilateral contracts coordinated by P1, not a multi-party contract. P1 holds the overall responsibility towards C; the sub-SLAs pass the relevant commitments on to P2 and P3 (back-to-back). P1 MUST keep the SLA Master consistent with the sub-SLAs (same SLI definitions and metrics, compatible targets and validity periods); in the demonstrator P1 does this manually with the providers. All contracts are held in P1's DCS; C, P2 and P3 log in with their own credentials. Because contracts may contain confidential or personal information, each organisation reads only the contracts to which it is a party (party read scoping, [DCS-FR-CWE-07]): C the SLA Master, P2 the sub-SLA P1↔P2, P3 the sub-SLA P1↔P3.

© 2026. This work is licensed under a CC BY 4.0 license.

2.5 Process overview

The demonstrator runs through a preparation step and four phases; step numbers refer to Figure 2.

Phase	Step	Activity
Preparation	7-8	FAP PCI issues PoA credentials to each organisation's contract signers (OID4VCI via OCM W-Stack); OCM issues entitlement credentials (DCS roles) to the users' PCM Cloud wallets.
I Selection	1	P2 and P3 author and publish their SLA sub-templates in their own baseline Federated Catalogue.
	2	Providers upload cloud service descriptions to CAT, validated against the Trusted Cloud service schema.
	3	FAP DCM harvests service descriptions (CAT) and SLA sub-templates (FC P2, FC P3) into its local catalogue.
	4	P1 searches and selects services and matching SLA sub-templates in the FAP DCM UI.
II Initiation	5	P1 creates the SLA Master template with its own commitments and imports the selected sub-templates into its DCS (template catalogue integration or ORCE import flow; own review and approval).
	6	P1 assembles the package (SLA Master C↔P1, sub-SLAs P1↔P2 and P1↔P3), invites the parties and assigns roles and parts.
III Negotiation	6	A party of a contract proposes a change; the other party confirms or modifies it; then review and approval.
IV Signing	9	Users of C, P1, P2 and P3 log in to P1's DCS and sign their own signature fields via OID4VP from PCM Cloud; signing is asynchronous.
	10	All contracts of the package are approved before the first signature. Sub-SLAs are preferably signed before the SLA Master (SM-03); a sub-SLA becomes ACTIVE only after the SLA Master is signed. When all contracts are ACTIVE, the package is complete and each party downloads its signed contracts.

2.6 Constraints

- The FACIS DCS core is used without change (Section 2.2).
- Contracting uses one DCS instance, operated by P1 and shared by all organisations; DCS-to-DCS communication ([DCS-FR-CWE-01]) is not used. A provider MAY operate its own DCS instance solely as an editor for SLA sub-templates (CAT-02).
- All components are deployed as demonstrator instances at TRL 4 (NFR-01).
- SLA content is machine-readable (JSON-LD, ODRL, SHACL) and follows the FACIS SLA Taxonomy; human-readable output is PDF/A-3. The SLA vocabulary (JSON-LD context and SHACL shapes for the Taxonomy) does not exist yet and is created in the demonstrator (TR-01).
- The demonstrator inherits the product constraints, operating environment, safety and security requirements of [FACIS.DCS.SRS] Sections 2.3, 2.5 and 3.3 except for production readiness, scalability and high availability (NFR-01).

© 2026. This work is licensed under a CC BY 4.0 license.

2.7 Dependencies

The demonstrator depends on the following capabilities of the DCS core and the XFSC components:

ID	Dependency	Affects
D-01	Import of SLA templates via the DCS template catalogue integration (/catalogue/template/search, import). An imported template is created as Draft and passes review and approval, with SHACL validation, in P1's DCS. The DCS catalogue integration connects to one configured Federated Catalogue; sub-templates from several baseline catalogues therefore reach P1's DCS either via that catalogue or via an ORCE flow that creates the template through the DCS template API. Sub-templates MUST be in the DCS template format.	CAT-05, TR-01
D-02	Contract creation inherits the full template content (clauses and ODRL) [DCS-FR-CWE-13]; contracts are linked as parent and child [DCS-FR-CWE-02].	CWE-01, CWE-02
D-03	The parties of a contract negotiate its content; each organisation reads only the contracts to which it is a party ([DCS-FR-CWE-07], [DCS-FR-CWE-18]).	CWE-04, CSA-01
D-04	Signing by several signers of a contract in any order [DCS-FR-SM-07], [DCS-FR-SM-17], with the PoA credential checked against the signature field [DCS-FR-SM-03] and a PAdES signature applied [DCS-FR-SM-16].	SM-01, SM-02
D-05	Submission for review [DCS-FR-CWE-14], activation on last signature [DCS-FR-CWE-06] and PDF/A-3 export [DCS-FR-SM-27].	CWE-07, SM-04, CSA-02
D-06	OID4VP login and signing with credentials held in PCM Cloud and issued by FAP PCI and OCM, with the issuers configured as trust anchors.	IAM-01 to IAM-04
D-07	A contract made from a SUB_AGREEMENT template is deployed (ACTIVE) only when its parent is SIGNED or ACTIVE (hierarchy rule of the deployment gate, [DCS-FR-TR-26]).	SM-03, SM-04

2.8 User documentation

- **End-user guide:** short product overview, participants and roles, step-by-step flows for the four phases, wallet set-up and troubleshooting.
- **Administrator guide:** installation and configuration of all components of Section 2.2, trust anchors, role mapping, demo data and reset procedure.
- **Mandatory content of the end-user guide:** (1) the scope of the demonstrator (lead-provider model); (2) that compliance-check results shown by the DCS are structural and technical rule checks, not a legal or regulatory validation (PACM-02); (3) that SLOs are harmonised manually between the SLA Master and the sub-SLAs; (4) that each party sees only the contracts to which it is a party; (5) the manual handling of a stalled signing process (SM-05).

© 2026. This work is licensed under a CC BY 4.0 license.

3 Functional Requirements

3.1 Phase I – Selection (discovery)

[SLA-DEMO-CAT-01] Publication of Cloud Service Descriptions

Priority	MUST
Phase / layer	I Selection – catalogue layer (step 2)
Realised by	Service editor of the demonstrator UI (ORC-03) on the CAT FC API
Relation to DCS baseline	None (outside DCS).
Description	Providers MUST be able to create, edit and publish their service descriptions in CAT. A service description describes the service itself; SLA commitments (SLOs with SLIs and metrics) are maintained only in the linked sub-template (CAT-02). The editor (Figure 3) shows a service list with the state of each entry (Draft or Published) and a properties panel: service name, service type, the mandatory properties of the Trusted Cloud schema (e.g. region), description, and the links to the provider's SLA sub-templates. SLO values shown in the editor are read from the linked sub-template. The panel offers the actions Cancel, Save (stores a Draft) and Publish. Publish MUST validate the description against the Trusted Cloud service description schema (SHACL / JSON-LD) and make it retrievable via the FC API only if the validation succeeds.
Acceptance criteria	1. P2 creates and publishes a service description with the editor; it is retrievable via the FC API. 2. Publishing without a mandatory property (e.g. region) is rejected with a message naming the missing property. 3. A description that violates the Trusted Cloud schema is rejected with a validation message. 4. A republished description replaces the previous one and is found by the next harvest (CAT-03).

© 2026. This work is licensed under a CC BY 4.0 license.

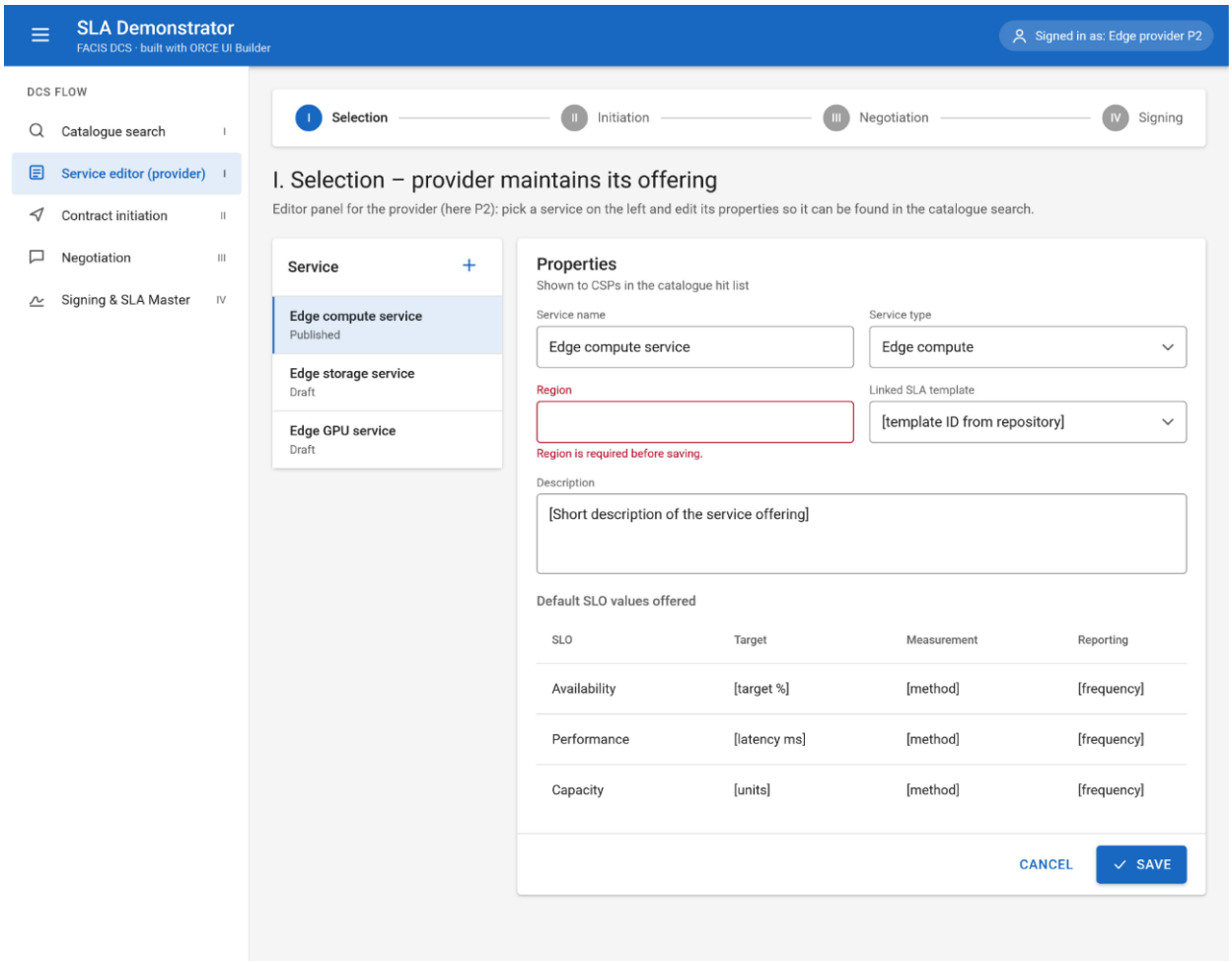


Figure 3: Screen sample – Service editor of the provider (Phase I); the SLO values shown are read from the linked SLA sub-template

[SLA-DEMO-CAT-02] Authoring and Publication of SLA Sub-Templates

Priority	MUST
Phase / layer	I Selection – catalogue layer (step 1)
Realised by	Baseline Federated Catalogue of P2 and P3 (asset store, FC API); authoring with the template editor of a provider-operated DCS instance and publication via [DCS-IR-SI-01], or with the SLA template UI of the demonstrator
Relation to DCS baseline	Prepares input for [DCS-FR-TR-01] Machine-Readable Format and [DCS-FR-TR-03] Semantic Hub.
Description	Each provider MUST be able to create, edit and publish its SLA sub-templates in its own baseline Federated Catalogue. A sub-template MUST contain the provider's SLOs with their SLIs and metrics, structured according to the FACIS SLA Taxonomy, and the monitoring, reporting and enforcement provisions that apply to the provider's part. It MUST reference the service description it applies to (covered service) and MUST NOT repeat its content. Sub-templates MUST be stored as versioned assets in the DCS template format (JSON-LD with dcs:documentStructure, dcs:contractFields and an ODRL policy) extended by the SLA vocabulary of TR-01, so that P1's DCS can import them without conversion.
Acceptance criteria	1. P2 and P3 each publish at least one SLA sub-template with at least two SLOs, each with SLI and metric.

© 2026. This work is licensed under a CC BY 4.0 license.

	2. A sub-template that does not conform to the SLA vocabulary of TR-01 (SHACL shapes) cannot be published. 3. Each published version remains retrievable by version.

[SLA-DEMO-CAT-03] Harvesting into FAP DCM

Priority	MUST
Phase / layer	I Selection (step 3)
Realised by	FAP DCM (catalogue registry, schema registry, asset crawler and mapper, local catalogue)
Relation to DCS baseline	None (outside DCS).
Description	FAP DCM MUST harvest service descriptions from CAT and SLA sub-templates from the baseline catalogues of P2 and P3 and store them in its local catalogue with harmonised metadata and provenance (source catalogue, provider, version).
Acceptance criteria	1. After a harvest run, all published service descriptions and sub-templates of P2 and P3 appear in the local catalogue. 2. Each entry shows its source catalogue, provider and version. 3. Each sub-template is linked to the service description it refers to; a service description can have several sub-templates (e.g. different service levels).

[SLA-DEMO-CAT-04] Search, Hit List and Selection

Priority	MUST
Phase / layer	I Selection (step 4)
Realised by	FAP DCM search and selection UI (ORCE UI Builder)
Relation to DCS baseline	Search is provided by FAP DCM; [DCS-FR-TR-10] is not used.
Description	P1 MUST be able to search the federated catalogues via FAP DCM. The search runs across all connected catalogues and returns one combined hit list; P1 filters it and selects the services and the matching SLA sub-templates (Figure 4). Search and filter fields: free text, service type, catalogue source, and the minimum value of an SLO taken from the linked sub-template (in the demonstrator: availability). Hit list columns: service, provider, catalogue, service type, whether an SLA sub-template exists, the SLO value with its SLI and metric, and the verification status of the service description. The hit list shows the number of candidates found and selected, supports paging and multiple selection, and offers the actions Reset, Search, Clear selection and Continue to initiation. SLO values of different providers are comparable only if they use the same SLI and metric. The hit list therefore shows the SLI and metric (measurement method) next to each value; the filter applies to the value only. Differing metrics are not harmonised by the demonstrator.
Acceptance criteria	1. A search by service type or free text returns one combined hit list from CAT and both provider catalogues. 2. The hit list can be filtered by service type, catalogue source and minimum availability. The SLI and metric used by each provider are shown next to the SLO value.

© 2026. This work is licensed under a CC BY 4.0 license.

	3. Candidates without an SLA sub-template are marked as such and are excluded when an SLO filter is set. 4. P1 selects services and sub-templates; Continue to initiation hands the selection over to the import (CAT-05).

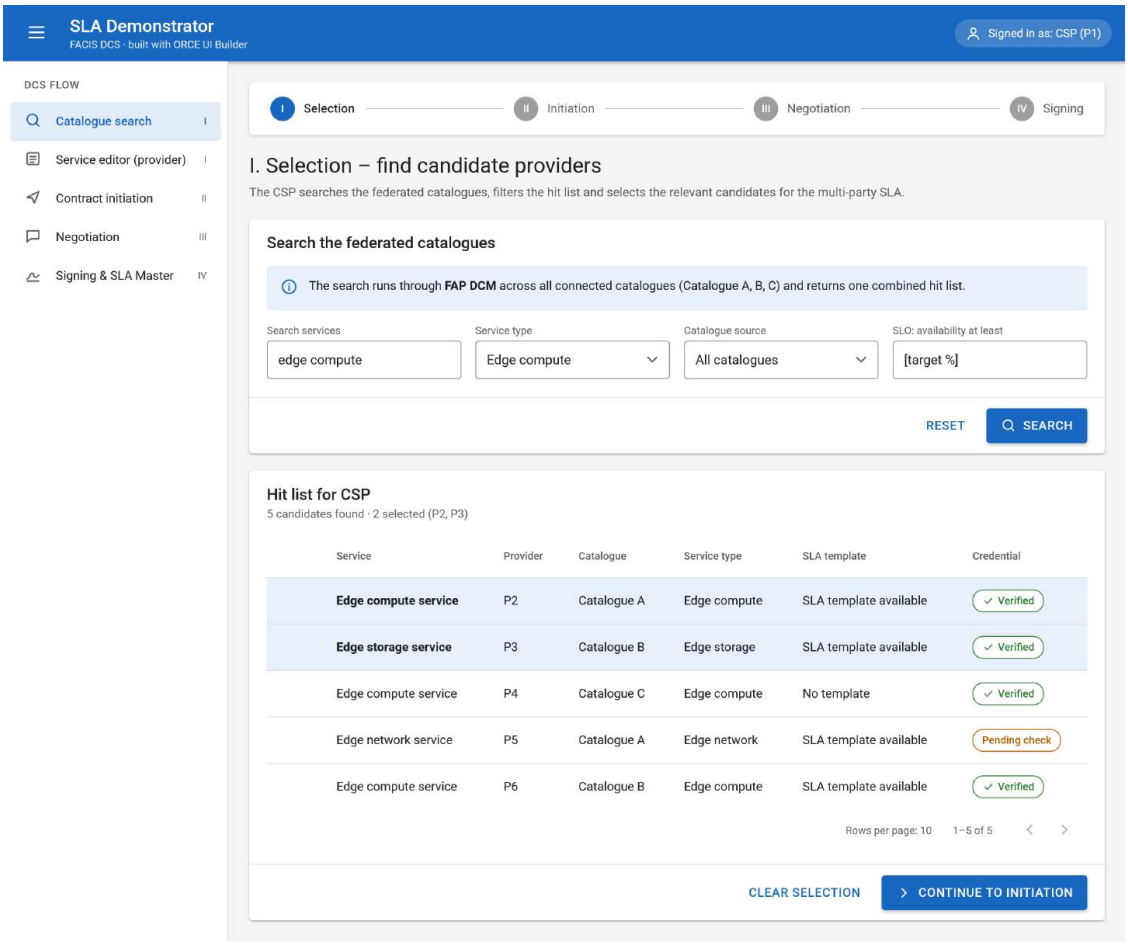


Figure 4: Screen sample – Catalogue search and hit list of P1 (Phase I)

[SLA-DEMO-CAT-05] Import of Selected Templates into the DCS

Priority	MUST
Phase / layer	II Initiation (step 5)
Realised by	DCS template catalogue integration and template repository; ORCE import flow where the sub-templates are not available in the catalogue connected to P1's DCS (D-01)
Relation to DCS baseline	Reuses [DCS-IR-SI-01] catalogue integration, [DCS-FR-TR-13] Template Creation, [DCS-FR-TR-14/15] review and approval, [DCS-FR-TR-03] Semantic Hub.
Description	P1 MUST be able to import the selected SLA sub-templates into its DCS; they are registered with the template level SUB_AGREEMENT. The SLA Master template is created by P1 in its DCS (TR-03). Imported templates pass the DCS review and approval (with SHACL validation) before they can be used for contract creation. The template provenance MUST record the source catalogue, provider and version.
Acceptance criteria	1. A selected sub-template of P2 is imported, approved by P1 and available for contract creation.

© 2026. This work is licensed under a CC BY 4.0 license.

	2. The provenance of the imported template shows the source catalogue, provider and version. 3. A template failing the SHACL validation is not approved.
Notes	Dependency: D-01.

3.2 SLA templates

[SLA-DEMO-TR-01] SLA Template Tree aligned with the FACIS SLA Taxonomy

Priority	MUST
Phase / layer	I Selection and II Initiation
Realised by	DCS template repository and Semantic Hub
Relation to DCS baseline	Reuses [DCS-FR-TR-01], [DCS-FR-TR-02] Multi-Tiered Contract Template Management and [DCS-FR-TR-04] machine- and human-readable linking.
Description	SLA templates MUST be structured as a tree: a main element (SLA Master template, DCS template level FRAME_AGREEMENT) with subordinate elements (provider sub-templates, level SUB_AGREEMENT). The main element MUST cover the building blocks Foundation, Service Level Objectives, Monitoring and Enforcement of the FACIS SLA Taxonomy; subordinate elements MUST contain the SLOs, SLIs and metrics of one provider and the monitoring, reporting and enforcement provisions (e.g. remedies, penalties) that apply to them. Every element MUST be machine-readable (JSON-LD / ODRL) and linked to a human-readable rendering. The demonstrator MUST provide the SLA vocabulary: a JSON-LD context and SHACL shapes for the FACIS SLA Taxonomy (building blocks, SLO → SLI → metric, responsible party), registered in the DCS Semantic Hub [DCS-FR-TR-03]. Starting points are the JSON template of the Taxonomy [FACIS-TAX], and the SLA template example of the DCS code (features/fixtures/sla_hosting_template.jsonld).
Acceptance criteria	1. The SLA Master template exposes addressable structures for the four building blocks. 2. Every SLO in the SLA Master template and in each sub-template has at least one SLI and one metric. 3. The template tree can be retrieved from the DCS and rendered human-readable. 4. The SLA vocabulary is registered in the Semantic Hub and validates the SLA Master template and the sub-templates.

© 2026. This work is licensed under a CC BY 4.0 license.

[SLA-DEMO-TR-02] Responsibility Assignment

Priority	MUST
Phase / layer	II Initiation
Realised by	SHACL shapes of the SLA vocabulary in the Semantic Hub
Relation to DCS baseline	Uses [DCS-FR-TR-03] Semantic Hub; does not rely on [DCS-FR-TR-23] or [DCS-FR-TR-26].
Description	Each SLO, SLI and metric MUST name the party or parties accountable for delivering it and the party responsible for measuring it, so that responsibilities are transparent to C and P1. Dependencies between service components and a liability model are not modelled.
Acceptance criteria	1. A template containing an SLO, SLI or metric without an accountable party or without a measuring party fails the SHACL validation. 2. The accountable and the measuring party are shown in the human-readable rendering.

[SLA-DEMO-TR-03] SLA Master Template

Priority	MUST
Phase / layer	II Initiation (step 5)
Realised by	DCS template repository (template editor, review and approval) and Semantic Hub
Relation to DCS baseline	Reuses [DCS-IR-TR-01] Template Builder, [DCS-FR-TR-02] Multi-Tiered Contract Template Management (level FRAME_AGREEMENT) and [DCS-FR-TR-14/15] submission, review and approval.
Description	P1 MUST be able to create the SLA Master template in its DCS with the template editor. The SLA Master template covers the four building blocks of the Taxonomy, P1's own service commitments (SLOs with SLIs and metrics for the service components that P1 delivers itself, e.g. the ERP hosting), the end-to-end SLOs towards C, and per SLO the responsible provider (TR-02).
Acceptance criteria	1. P1 creates an SLA Master template with at least one own SLO and one end-to-end SLO, each with SLI and metric. 2. The template passes the SHACL validation of the SLA vocabulary, review and approval in P1's DCS and is available for contract creation.

3.3 Phase II – Contract initiation

The demonstrator implements the initiation path driven by the lead provider: P1 creates the SLA Master template, selects the sub-providers and initiates the contracts.

[SLA-DEMO-CWE-01] Selection of the SLA Template

Priority	MUST
Phase / layer	II Initiation (step 6)
Realised by	DCS contract workflow engine
Relation to DCS baseline	Reuses [DCS-FR-CWE-13] Contract Creation.

© 2026. This work is licensed under a CC BY 4.0 license.

Description	P1 MUST be able to select the SLA Master template (TR-03) and the imported sub-templates from the repository and create the contracts from them. Metadata of the parties (name, identifier) MUST be filled in automatically from their credentials.
Acceptance criteria	1. P1 creates the SLA Master and the sub-SLAs from approved templates; each contract contains the full template content. 2. Party name and identifier are filled in from the credentials without manual input.
Notes	Dependency: D-02.

[SLA-DEMO-CWE-02] Assembly of the Contract Package

Priority	MUST
Phase / layer	II Initiation (step 6)
Realised by	DCS contract workflow engine
Relation to DCS baseline	Reuses [DCS-FR-CWE-02] Hierarchical Contract Structures (parentContract) and the two-party model of [DCS-FR-CWE-01].
Description	P1 MUST be able to assemble a contract package consisting of the SLA Master C↔P1 and one sub-SLA per provider (P1↔P2, P1↔P3), each sub-SLA linked to the SLA Master as child contract. The package MUST be extendable by further sub-SLAs (P4–Pn) following the same pattern. The SLA Master lists the attached sub-SLAs with provider and covered service, so that C knows the sub-providers when it reviews the SLA Master and can reject a sub-provider by a change proposal (CWE-05).
Acceptance criteria	1. A package with one SLA Master and two sub-SLAs is created; the hierarchy is visible in the DCS. 2. A third sub-SLA for a further provider can be added to the package before the first signature without changes to the DCS configuration or code. 3. The human-readable export of the SLA Master lists the attached sub-SLAs and their providers.

[SLA-DEMO-CWE-03] Invitation of Parties and Role Assignment

Priority	MUST
Phase / layer	II Initiation (step 6)
Realised by	DCS contract workflow engine; entitlement credentials (OCM)
Relation to DCS baseline	Reuses [DCS-FR-CWE-07] Role-Based Access Control for the Contract Workflow Engine.
Description	P1 MUST be able to invite the other party of each contract (C, P2 or P3) and assign roles (reviewer, approver, signer) to its users. The initiation view (Figure 5) shows a party and role table and the actions Add party and Send to; Send to names the invited parties. Invited users access the contract in P1's DCS with their own credentials.
Acceptance criteria	1. P1 invites P2 to the sub-SLA P1↔P2 and assigns reviewer, approver and signer roles; a user of P2 with matching entitlement can open the contract. 2. A user without an assigned role cannot open the contract.

© 2026. This work is licensed under a CC BY 4.0 license.

[SLA-DEMO-CWE-04] Part Assignment and Access

Priority	MUST
Phase / layer	II Initiation
Realised by	SLA template content (responsible party per element, TR-02) and the party read scoping of the DCS
Relation to DCS baseline	Reuses the party read scoping of [DCS-FR-CWE-07] and [DCS-FR-CSA-02]; field-level edit rights per party are not used.
Description	Each SLA element (SLO, SLI, metric) MUST be assigned to one or more responsible providers (P1, P2, P3) via an “Assigned to” selection in the template tree (Figure 5). An element assigned to P2 or P3 is included in the sub-SLA of each assigned provider; P1’s own elements and the end-to-end SLOs remain in the SLA Master. Each organisation reads and negotiates only the contracts to which it is a party (Section 2.4); P2 and P3 have no access to the SLA Master or to each other’s sub-SLA. P1 MAY share selected content with a provider outside the DCS, e.g. as an export.
Acceptance criteria	1. P2 can open and negotiate the sub-SLA P1↔P2 and cannot open the SLA Master or the sub-SLA P1↔P3. 2. An element assigned to P2 appears in the sub-SLA P1↔P2; an element assigned to P1 remains in the SLA Master. 3. C can open and negotiate the SLA Master, which shows the responsible provider per SLO and the attached sub-SLAs; C cannot open the sub-SLAs.
Notes	Dependency: D-03.

© 2026. This work is licensed under a CC BY 4.0 license.

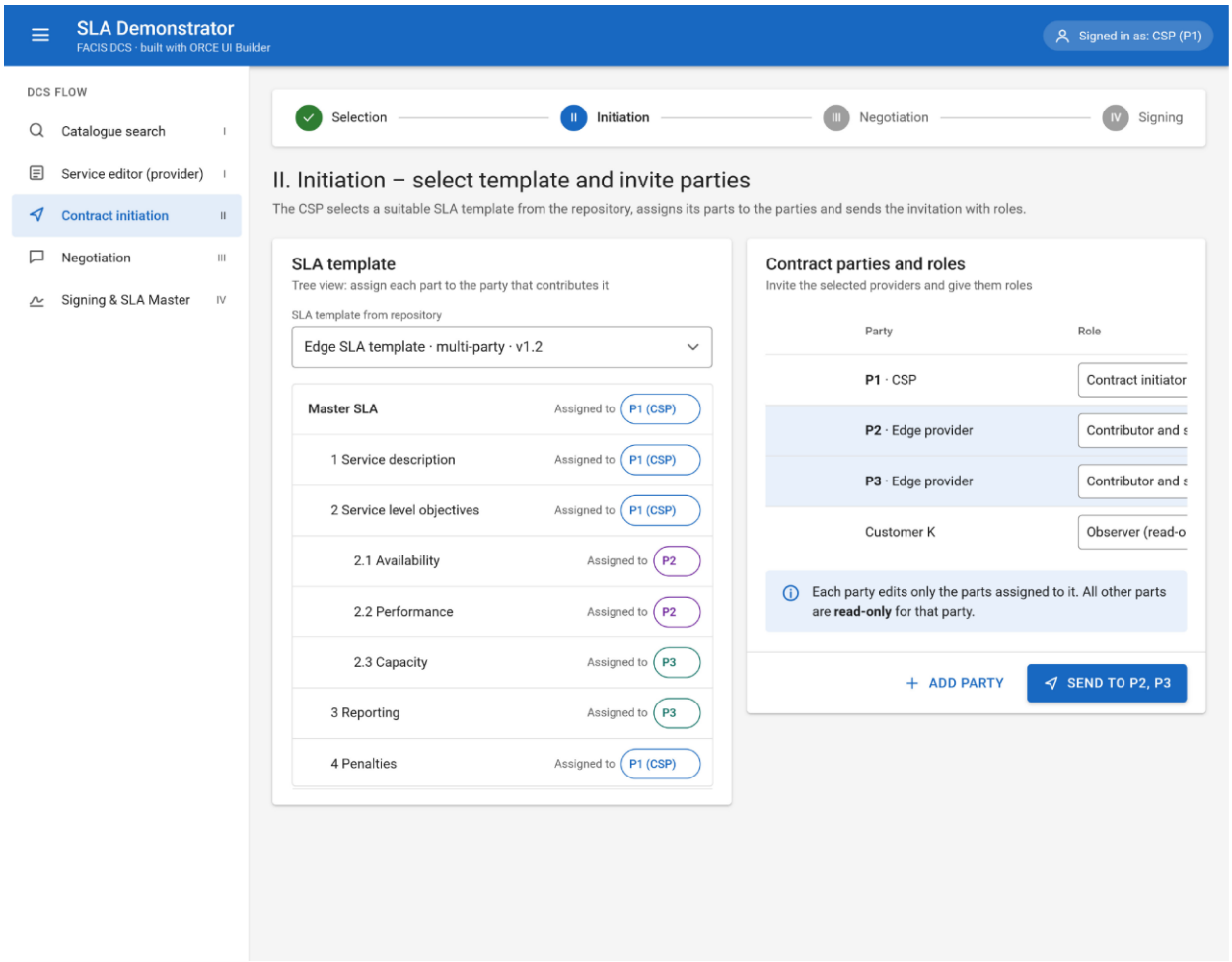


Figure 5: Screen sample – Contract initiation: SLA template tree with part assignment, contract parties and roles; read and edit rights follow the contracts as described in CWE-04

3.4 Phase III – Contract negotiation (simplified)

[SLA-DEMO-CWE-05] Change of Text Elements

Priority	MUST
Phase / layer	III Negotiation
Realised by	DCS contract workflow engine
Relation to DCS baseline	Reuses [DCS-FR-CWE-18] Contract Negotiation and contract versioning.
Description	Each party of a contract MUST be able to propose a change to a text element of that contract (redline proposal per [DCS-FR-CWE-18]). Every change creates a new contract version.
Acceptance criteria	1. P1 proposes a new target value for an SLO in the sub-SLA P1↔P2; a new version is created and the previous version remains retrievable. 2. C proposes a change to the SLA Master; P1 receives it for confirmation.

© 2026. This work is licensed under a CC BY 4.0 license.

[SLA-DEMO-CWE-06] Confirmation or Modification by the Other Party

Priority	MUST
Phase / layer	III Negotiation
Realised by	DCS contract workflow engine
Relation to DCS baseline	Reuses [DCS-FR-CWE-18] Contract Negotiation and [DCS-FR-CWE-25] review interface (negotiation threads).
Description	The other party of the contract MUST confirm or modify the proposed change. A modification creates a new version, which the proposing party in turn confirms or modifies. Consistency with the other contracts of the package is ensured manually by P1 (Section 2.4). The negotiation view (Figure 6) shows the SLA structure with editable and read-only parts and a “Changed” marker, the changed text element with previous and new value, the assigned party and the proposing party, an optional comment, the actions Modify and Confirm, read-only parts of other parties, and the responses per party (Proposed, Pending, Confirmed). The action Go to signing is enabled only when both parties have confirmed all changes and the contract is approved (CWE-07).
Acceptance criteria	1. P2 confirms a change of P1; the contract can proceed to approval. 2. P2 modifies the change; P1 sees the new version and confirms it. 3. The response status of each party is shown for every change. 4. All confirmations, modifications and comments are recorded with actor and time.

© 2026. This work is licensed under a CC BY 4.0 license.

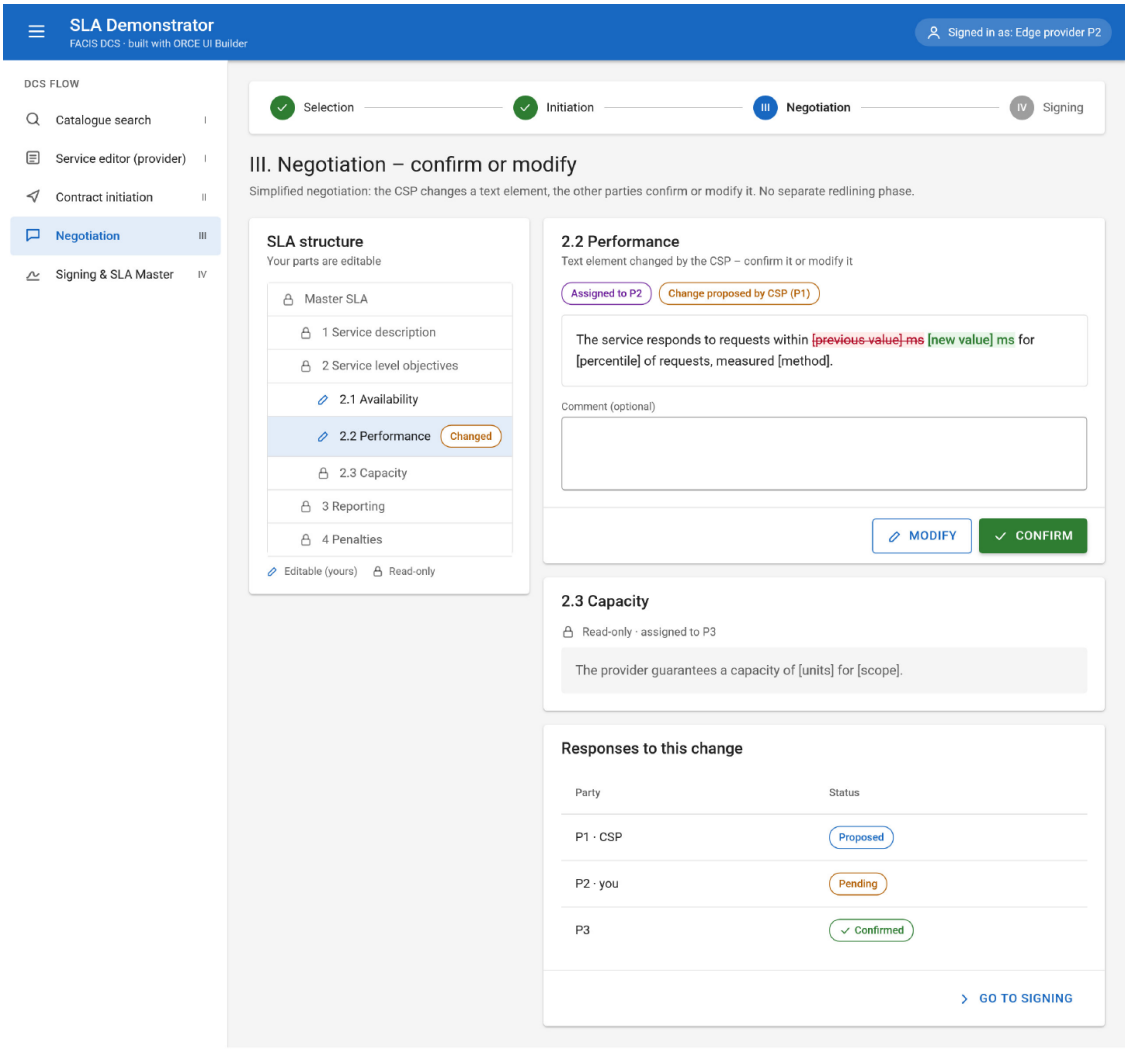


Figure 6: Screen sample – Negotiation: confirm or modify a changed text element (Phase III)

[SLA-DEMO-CWE-07] Review and Approval

Priority	MUST
Phase / layer	III Negotiation
Realised by	DCS contract workflow engine
Relation to DCS baseline	Reuses [DCS-FR-CWE-14] Contract Submission for Review and [DCS-FR-CWE-15] Contract Review and Approval.
Description	When both parties of a contract have confirmed all changes, P1 MUST submit the contract for review and approval. Only contracts approved by both parties can be signed.
Acceptance criteria	1. A contract with an unconfirmed change cannot be approved. 2. After approval by both parties, the contract is released for signing.
Notes	Dependency: D-05.

© 2026. This work is licensed under a CC BY 4.0 license.

3.5 Phase IV – Signing

[SLA-DEMO-SM-01] Invitation of PoA Holders to Sign

Priority	MUST
Phase / layer	IV Signing (step 9)
Realised by	DCS signature management; PoA credentials from FAP PCI; PCM Cloud
Relation to DCS baseline	Reuses [DCS-FR-SM-03] Signing Identity and PoA Authorization Credentials, [DCS-FR-SM-07] Multi-Signature and Role-Based Signing Flows, [DCS-FR-SM-16] Apply Digital Signature and [DCS-FR-SM-23] wallet-based signing.
Description	Users holding a PoA credential for their organisation MUST be invited to sign the contracts to which their organisation is a party. In the signing ceremony, the signer presents the PoA credential via OID4VP from PCM Cloud; the DCS checks the PoA against the signature field and applies an advanced electronic signature (PAdES).
Acceptance criteria	1. A signer of P2 with a valid PoA signs the signature field of P2 in the sub-SLA P1↔P2. 2. A user without a valid PoA for the organisation cannot sign; the attempt is logged. 3. Each signature can be verified individually.
Notes	Dependency: D-04.

[SLA-DEMO-SM-02] Asynchronous Signing

Priority	MUST
Phase / layer	IV Signing
Realised by	DCS signature management
Relation to DCS baseline	Reuses [DCS-FR-SM-07] (signing in any order) and [DCS-FR-SM-17] Multi-Signer Support.
Description	Signing MUST be asynchronous: the signers of a contract sign independently, at different times and in any order. The signing status MUST be visible per signer (at least Pending and Signed). A partly signed contract is not concluded and MUST NOT be deployed. The signing view (Figure 7) shows per party the signer (PoA holder), the PoA credential status, the invitation date and the signing status (Pending, Signed), a counter of completed signatures, and the action Resend invitation. A side panel shows the SLA Master with its identifier (DID or UUID), the SPOC and the status of the SLA Master and of each sub-SLA.
Acceptance criteria	1. C and P1 sign the SLA Master on different days and in either order. 2. The status per signer is visible to all parties of the contract. 3. P1 can resend the invitation to a signer who has not yet signed. 4. A partly signed contract is neither archived as executed nor deployed.

© 2026. This work is licensed under a CC BY 4.0 license.

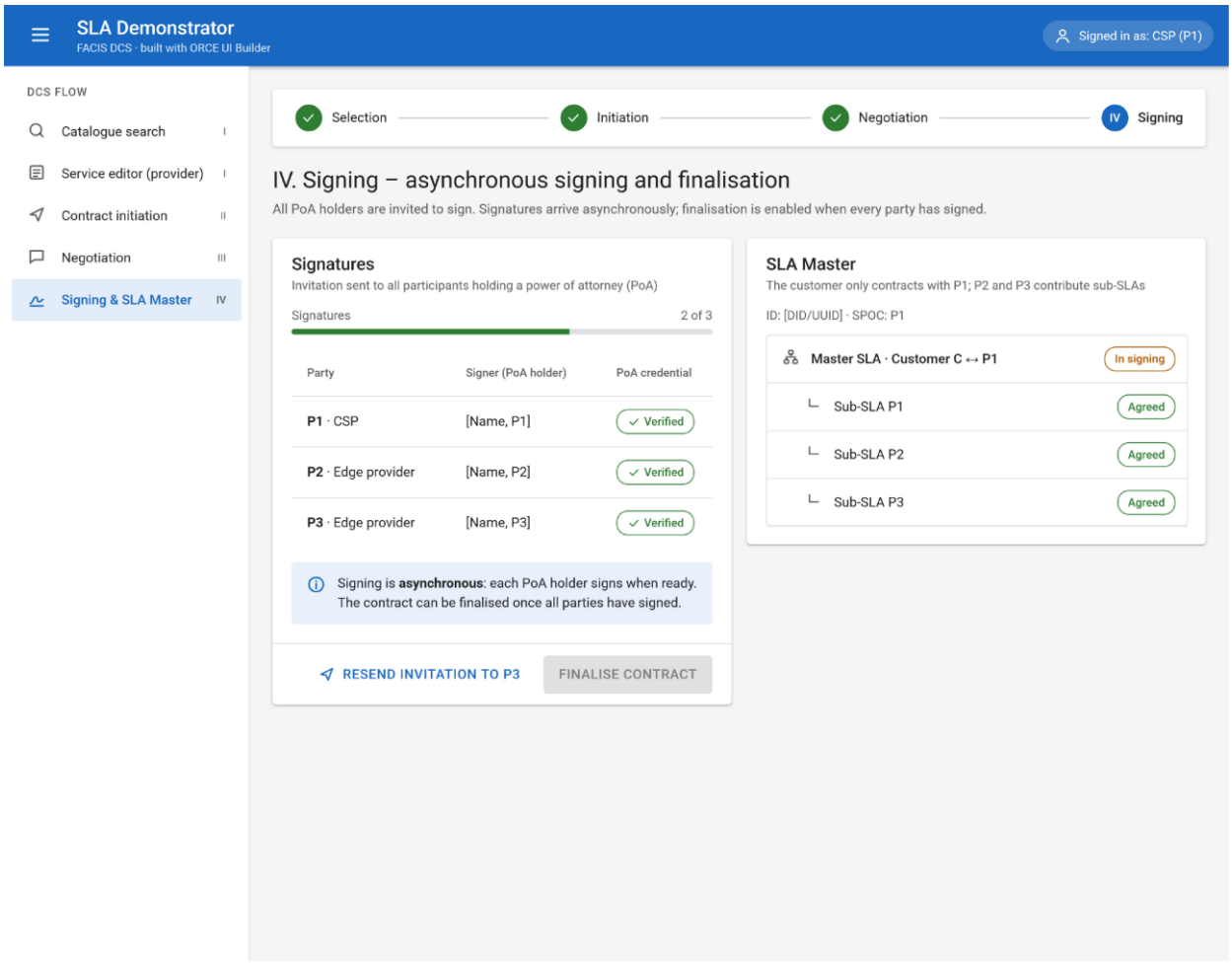


Figure 7: Screen sample – Signing and SLA Master overview (Phase IV)

[SLA-DEMO-SM-03] Signing Sequence across the Package

Priority	MUST
Phase / layer	IV Signing (step 10)
Realised by	Hierarchy rule of the DCS deployment gate (D-07); condition clause in the sub-SLA templates; deployment of the sub-SLAs triggered by P1 or by an ORCE flow on DCS webhooks (ORC-01)
Relation to DCS baseline	No DCS change; uses DCS workflow events and the hierarchy rule of [DCS-FR-TR-26] (D-07).
Description	All contracts of the package MUST be approved before the first signature. The sub-SLAs SHOULD be signed before the SLA Master, so that P1 holds the back-to-back commitments of P2 and P3 before it commits to C. Each sub-SLA MUST be conditional on the SLA Master, so that P1 is not bound to P2 and P3 if C does not sign: the sub-SLA template contains a clause that the sub-SLA takes effect only when the SLA Master is signed, and the DCS sets a sub-SLA to ACTIVE only when its parent SLA Master is SIGNED or ACTIVE (D-07).
Acceptance criteria	1. A signed sub-SLA is not set to ACTIVE while the SLA Master is not signed. 2. After the SLA Master is signed, the signed sub-SLAs are deployed and set to ACTIVE (triggered by P1 or by the ORCE flow of ORC-01). 3. The sub-SLA templates contain the clause that makes the sub-SLA conditional on the SLA Master.

© 2026. This work is licensed under a CC BY 4.0 license.

[SLA-DEMO-SM-04] Finalisation and Activation

Priority	MUST
Phase / layer	IV Signing (step 10)
Realised by	DCS signature management, contract storage and archive, deployment
Relation to DCS baseline	Reuses [DCS-FR-SM-12] Contract Deployment Trigger, [DCS-FR-CSA-08] Store Signed Contract in Archive and [DCS-FR-CWE-06] Event-Driven Contract Execution.
Description	After the last signature of a contract, the DCS MUST archive the contract as signed PDF/A-3 with embedded machine-readable content and set it to ACTIVE; for a sub-SLA, activation is subject to SM-03. The package is complete when all its contracts are ACTIVE. The action Finalise package in the signing view is enabled only then; it confirms the completion and does not change any contract.
Acceptance criteria	1. After its last signature, a contract is archived; the SLA Master is set to ACTIVE immediately, a sub-SLA as defined in SM-03. 2. The archived PDF/A-3 contains all signatures and the machine-readable SLA.

[SLA-DEMO-SM-05] Stalled Signing and Revocation

Priority	SHOULD
Phase / layer	IV Signing
Realised by	Organisational procedure; DCS core functions
Relation to DCS baseline	Reuses [DCS-FR-SM-20] Signature Revocation; does not rely on [DCS-FR-SM-13].
Description	If the signing of a contract cannot be completed, P1 SHOULD withdraw or terminate the contract and, where needed, create a new version. A signature revocation by an authorised role invalidates the whole contract until it is signed again. Because each contract of the package is a separate two-party contract, a revocation affects only that contract; the other contracts remain valid. P1 resolves the consequences for the package organisationally (e.g. an SLO of the SLA Master that depends on a revoked sub-SLA, or sub-SLAs whose SLA Master is revoked). Deadlines, reminders, decline actions and automatic revocation are out of scope (Section 1.3.2).
Acceptance criteria	1. P1 can terminate a partly signed contract; the action is recorded in the audit trail. 2. A revoked contract shows the state REVOKED; signing again restores validity as a new version.

3.6 Storage, export and audit

[SLA-DEMO-CSA-01] Archive of the Contract Package

Priority	MUST
Phase / layer	IV and after
Realised by	DCS contract storage and archive
Relation to DCS baseline	Reuses [DCS-FR-CSA-08] and [DCS-FR-CSA-02] Role-Based Access Control.

© 2026. This work is licensed under a CC BY 4.0 license.

Description	All signed contracts of the package MUST be archived in P1's DCS with their hierarchy. Each organisation MUST be able to read the archived contracts to which it is a party with its own credentials. After the package is complete (SM-04), each organisation MUST be able to download the signed PDF/A-3 and the audit trail of each of these contracts, so that it holds its own evidence independently of P1's DCS. The files MAY be delivered by an ORCE flow.
Acceptance criteria	1. C, P2 and P3 retrieve the archived contracts to which they are a party after logging in to P1's DCS. 2. An organisation that is not a party to a contract has no access to it. 3. C downloads the signed SLA Master as PDF/A-3 with its audit trail; P2 downloads the signed sub-SLA P1↔P2 with its audit trail.

[SLA-DEMO-CSA-02] Human- and Machine-Readable Export

Priority	MUST
Phase / layer	All phases
Realised by	DCS
Relation to DCS baseline	Reuses [DCS-FR-CWE-04], [DCS-FR-CWE-05] and [DCS-FR-SM-27] Support for PDF/A Format.
Description	At every state, each contract MUST be exportable in a human-readable form synchronised with the machine-readable representation. The export of the package MUST show which provider is responsible for which SLO, SLI and metric. A package export for a party contains only the contracts that this party may read ([DCS-FR-CWE-30]).
Acceptance criteria	1. The human-readable and machine-readable exports of a contract have the same content. 2. The package export labels the responsible provider per SLO. 3. The signed PDF/A-3 passes an independent conformance check (e.g. veraPDF).

[SLA-DEMO-PACM-01] Audit Trail

Priority	MUST
Phase / layer	All phases
Realised by	DCS process audit; FAP DCM provenance
Relation to DCS baseline	Reuses [DCS-FR-PACM-01] Tamper-Proof Audit Trail for Contract Lifecycle.
Description	The actions of every actor MUST be recorded: template import and approval, contract creation, invitations, changes, confirmations and modifications, approvals, signatures, revocations and terminations. The origin of imported templates is covered by their provenance.
Acceptance criteria	1. The audit trail of a contract shows a chronological, per-actor record of all lifecycle events. 2. Audit entries are immutable and exportable.

© 2026. This work is licensed under a CC BY 4.0 license.

[SLA-DEMO-PACM-02] Clarification of Compliance Checks

Priority	MUST
Phase / layer	All phases
Realised by	User documentation
Relation to DCS baseline	Clarifies [DCS-FR-TR-07] and [DCS-FR-PACM-03]; DCS labels remain unchanged.
Description	The demonstrator MUST NOT be presented as performing legal or regulatory compliance validation of SLA content. The end-user guide MUST state (Section 2.8) that compliance findings in the DCS are structural and technical rule checks.
Acceptance criteria	1. The end-user guide contains the statement on the meaning of compliance findings. 2. No contract is approved or blocked on the basis of a legal or regulatory verdict.

3.7 Identity, credentials and entitlement

[SLA-DEMO-IAM-01] PoA Issuance for Contract Signers

Priority	MUST
Phase / layer	Preparation (step 7)
Realised by	FAP PCI (tenant admin, issuer page, issuer plugin, ORCE issuance flow)
Relation to DCS baseline	Provides the credentials required by [DCS-FR-SM-03].
Description	Each organisation (C, P1, P2, P3) MUST be able to issue PoA credentials to its contract signers. The tenant administrator configures the organisation (OIDC, branding, credential types); the signer receives the PoA credential (organisation, role such as Contract Signer, signature, status entry) via OIDC login, ORCE issuance flow and QR code into PCM Cloud (OID4VCI via OCM W-Stack).
Acceptance criteria	1. A signer of each organisation holds a valid PoA credential in PCM Cloud. 2. A PoA credential can be revoked via the status list.

[SLA-DEMO-IAM-02] Entitlement Credentials for DCS Roles

Priority	MUST
Phase / layer	Preparation (step 8)
Realised by	OCM W-Stack
Relation to DCS baseline	Provides the role credentials for [DCS-FR-CWE-07], [DCS-FR-TR-06] and [DCS-FR-CSA-02].
Description	OCM MUST issue entitlement credentials that map to DCS roles to the users' PCM Cloud wallets (credential offer to wallet, OID4VCI) and verify their presentation (OID4VP).
Acceptance criteria	1. Users of C, P1, P2 and P3 receive entitlement credentials for their DCS roles. 2. A revoked entitlement credential no longer grants access.

© 2026. This work is licensed under a CC BY 4.0 license.

[SLA-DEMO-IAM-03] Personal Wallet and Login

Priority	MUST
Phase / layer	All phases
Realised by	PCM Cloud
Relation to DCS baseline	Uses the credential-based authentication of the DCS core (OID4VP).
Description	Each user of C, P1, P2 and P3 MUST have a PCM Cloud account that receives, holds and presents the PoA and entitlement credentials. Users log in to P1's DCS and take part in signing ceremonies by presenting their credentials via OID4VP.
Acceptance criteria	1. A user logs in to the DCS by presenting the entitlement credential from PCM Cloud. 2. The same user presents the PoA credential in the signing ceremony.

[SLA-DEMO-IAM-04] Trust Anchors

Priority	MUST
Phase / layer	Preparation
Realised by	OCM W-Stack; DCS configuration
Relation to DCS baseline	Configuration of the DCS core
Description	The issuers of C, P1, P2 and P3 MUST be configured as trusted issuers in P1's DCS for login and signing. Credentials from untrusted issuers MUST be rejected. The list of trusted issuers is agreed by all participants (ecosystem trust list, e.g. via the OCM trust anchors) and configured by P1.
Acceptance criteria	1. Credentials issued by the configured issuers of C, P1, P2 and P3 are accepted. 2. A credential from an unconfigured issuer is rejected and the attempt is logged.
Notes	Dependency: D-06.

3.8 Orchestration and extensibility

[SLA-DEMO-ORC-01] ORCE Flows

Priority	SHOULD
Phase / layer	All phases
Realised by	XFSC ORCE
Relation to DCS baseline	Uses DCS workflow webhooks and contract target flows; no DCS change.
Description	Functions not provided by the components of Section 2.2 SHOULD be realised as ORCE flows, in particular the deployment of the signed sub-SLAs once the SLA Master is signed (SM-03) and notifications to parties on changes and signing invitations.
Acceptance criteria	1. The deployment of the signed sub-SLAs can be triggered by an ORCE flow on the signing event of the SLA Master. 2. Parties are notified of a change requiring their confirmation.

© 2026. This work is licensed under a CC BY 4.0 license.

[SLA-DEMO-ORC-02] Additional Providers

Priority	SHOULD
Phase / layer	All phases
Realised by	Configuration of all components
Relation to DCS baseline	No DCS change.
Description	A further provider (P4–Pn) SHOULD be added by configuration only: a baseline catalogue, registration in FAP DCM, PoA and entitlement credentials, trust anchor, and an additional sub-SLA in the package.
Acceptance criteria	1. A fourth provider is added to an existing scenario and signs its sub-SLA without code changes.

[SLA-DEMO-ORC-03] Demonstrator Flow UI

Priority	MUST
Phase / layer	All phases
Realised by	XFSC ORCE UI Builder on the APIs of DCS, FAP DCM and the Federated Catalogues
Relation to DCS baseline	Uses the DCS APIs; no DCS change.
Description	The demonstrator MUST provide one web user interface that guides the participants through the four phases (Figures 3 to 7). It shows the signed-in participant and role, a navigation “DCS Flow” with the entries Catalogue search, Service editor (provider), Contract initiation, Negotiation, and Signing & SLA Master, each marked with its phase, and a phase stepper (Selection, Initiation, Negotiation, Signing) that marks completed phases. Only the functions permitted for the role of the signed-in participant are offered.
Acceptance criteria	1. Each participant (C, P1, P2, P3) signs in and sees its identity and role. 2. The stepper shows the current phase and marks completed phases. 3. A provider user sees the service editor and its negotiation parts but not the catalogue search of P1.

© 2026. This work is licensed under a CC BY 4.0 license.

4 Non-Functional Requirements

ID	Requirement	Priority
SLA-DEMO-NFR-01	TRL 4: all components run together in a laboratory or test environment and demonstrate the full process of Section 2.5 with demo data. Production readiness, scalability and high availability are not required.	MUST
SLA-DEMO-NFR-02	Usability: the provider and P1 interfaces are web UIs (editor, search and filter fields, hit list, selection) that can be used with the end-user guide as the only training.	MUST
SLA-DEMO-NFR-03	Security: credential-based authentication, role-based access control and transport security of the DCS core and the XFSC components are used unchanged.	MUST
SLA-DEMO-NFR-04	Reproducibility: the demonstrator can be reset to a defined initial state with demo organisations, users, credentials, catalogue entries and templates.	SHOULD
SLA-DEMO-NFR-05	Openness: configuration, schemas, templates and ORCE flows created for the demonstrator are published in the open-source repositories of the respective components.	SHOULD

5 Demonstration Scenario

Cloud service provider P1 hosts the ERP system of customer C. C requests edge capacity at two production sites. P1 cannot provide it alone and uses edge providers P2 and P3. P1 offers C one SLA Master that contains P1's own commitments for the ERP hosting and the end-to-end commitments for the edge sites, and to which the bilateral sub-SLAs with P2 and P3 are attached (lead-provider model). The demonstration runs as follows:

1. Tenant administrators issue PoA credentials; OCM issues entitlement credentials; all users hold their credentials in PCM Cloud.
2. P2 and P3 publish service descriptions in CAT and SLA sub-templates (e.g. availability ≥ 99.9 % per month, preprocessing latency ≤ 50 ms) in their baseline catalogues.
3. FAP DCM harvests the catalogues; P1 searches for edge services, filters the hit list and selects P2 and P3 with their sub-templates.
4. P1 creates the SLA Master template with its own SLOs, imports the sub-templates into its DCS, approves them and assembles the package: SLA Master C↔P1, sub-SLAs P1↔P2 and P1↔P3.
5. P1 invites C, P2 and P3 and assigns roles and parts; each party sees only the contracts to which it is a party.
6. P1 proposes a new availability target in the SLA Master; C confirms it. P2 proposes a change of a metric in its sub-SLA; P1 confirms it and checks manually that the SLA Master remains consistent.
7. The contracts are approved. P2, P3 and P1 sign the sub-SLAs on different days; then C and P1 sign the SLA Master. The sub-SLAs become ACTIVE only after the SLA Master is signed.
8. Each contract is archived as PDF/A-3 after its last signature; when all contracts are ACTIVE, the package is complete and each party downloads its signed contracts and their audit trails.

© 2026. This work is licensed under a CC BY 4.0 license.