

Supplementary Material:

FL-EHDS: A Privacy-Preserving Federated Learning Framework for the European Health Data Space

Fabio Liberti

Department of Computer Science
Universitas Mercatorum, Rome, Italy
fabio.liberti@unimercatorum.it
ORCID: 0000-0003-3019-5411

Abstract—This document provides supplementary material for the FL-EHDS paper, including complete algorithm pseudocode for all framework components, extended experimental figures, detailed algorithm comparison analysis, advanced FL paradigm descriptions, infrastructure component specifications, extended EHDS interoperability details, and clinical imaging experiment configurations. The open-source reference implementation (~40K lines, 159 modules) is available at <https://github.com/FabioLiberti/FL-EHDS-FLICS2026>.

I. PRISMA FLOW DIAGRAM

II. ALGORITHM PSEUDOCODE

This section provides formal algorithmic descriptions of all FL-EHDS framework components. Each algorithm is presented with: (1) a contextual explanation of *why* the component is needed in the EHDS regulatory context; (2) the formal pseudocode; and (3) practical considerations for deployment. The algorithms are organized following the data flow through the three-layer architecture: governance validation (Layer 1), privacy-preserving aggregation (Layer 2), and local data processing (Layer 3).

Reading guide: Algorithms S1–S4 form the core FL-EHDS training pipeline. Algorithms S5–S6 address EHDS-specific challenges (non-IID data and citizen opt-out). Algorithms S7–S8 handle data preprocessing and privacy budget management.

A. FedAvg with EHDS Compliance

Algorithm S1 presents the core federated averaging procedure adapted for EHDS regulatory requirements, operating in a client-server architecture where the central aggregator coordinates training across distributed hospital nodes within a Secure Processing Environment.

Key Design Decisions:

- **ValidatePermit:** Before each round, the HDAB-issued permit is verified against temporal bounds and Article 53 permitted purposes.
- **SelectParticipants:** Configurable client selection—full participation or sampling for large federations.
- **FilterOptedOut:** Records from citizens who exercised Article 71 opt-out rights are excluded *before* gradient computation.

- **Weighted Aggregation:** Gradients weighted by local dataset size (n_h), following original FedAvg [13].
- **ClipGradient:** L2-norm clipping bounds individual contributions, providing sensitivity bounds for DP.

Relationship to subsequent components: The ClipGradient operation in Algorithm S1 establishes a bounded sensitivity C for each client’s contribution. This bound is the prerequisite for Algorithm S2 (Gaussian DP), which calibrates noise proportional to C . Meanwhile, ValidatePermit invokes Algorithm S3 (Permit Validation) and FilterOptedOut invokes Algorithm S6 (Opt-Out Filtering).

B. Gaussian Differential Privacy Mechanism

Algorithm S2 implements the Gaussian mechanism for differential privacy, applied at the aggregation server after receiving clipped gradients.

Mathematical Foundation: The noise scale $\sigma = C \cdot \sqrt{2 \ln(1.25/\delta)}/\epsilon$ guarantees (ϵ, δ) -DP. The cumulative privacy expenditure is tracked using Rényi DP (RDP) [26] composition, providing 5–6× tighter bounds than naive composition.

Practical Considerations:

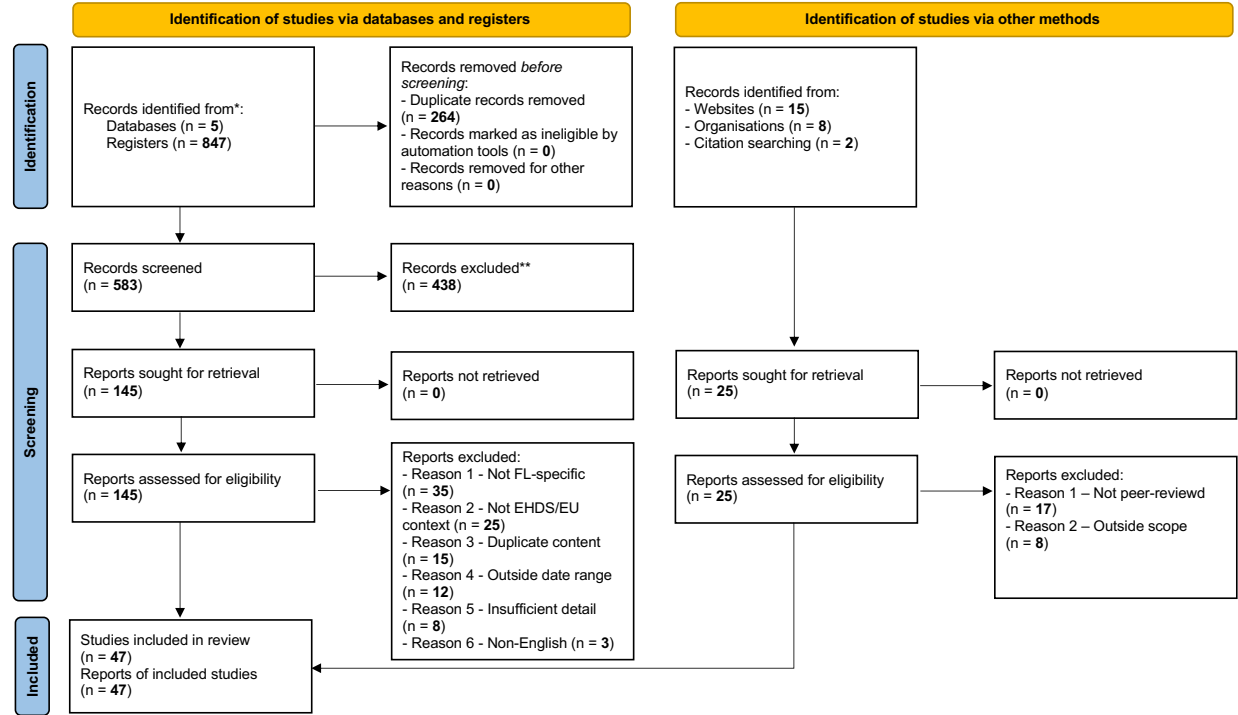
- $\epsilon = 10$: moderate noise, expected accuracy drop in the 5–6pp range (consistent with Wei et al. [27])
- $\epsilon = 1$: strong privacy, larger accuracy cost; a comprehensive ablation across ϵ values is planned
- The ϵ selection must be negotiated with HDABs during permit approval

Integration with privacy budget: The ϵ consumed by Algorithm S2 in each round is tracked by Algorithm S8 (RDP Privacy Budget Accountant). If the cumulative budget exceeds the threshold approved in the HDAB data permit, training is automatically terminated. The next algorithm (S3) formalizes the permit validation that authorizes each round.

C. HDAB Permit Validation

Algorithm S3 ensures all FL operations comply with the data permit issued by HDABs. Under EHDS Article 53, secondary use of health data is only lawful for specifically

PRISMA 2020 flow diagram for new systematic reviews which included searches of databases, registers and other sources



*Consider, if feasible to do so, reporting the number of records identified from each database or register searched (rather than the total number across all databases/registers).
**If automation tools were used, indicate how many records were excluded by a human and how many were excluded by automation tools.

Source: Page MJ, et al. BMJ 2021;372:n71. doi: 10.1136/bmj.n71.

This work is licensed under CC BY 4.0. To view a copy of this license, visit <https://creativecommons.org/licenses/by/4.0/>

Fig. 1. PRISMA 2020 flow diagram for the systematic review. Database searches across PubMed, IEEE Xplore, Scopus, Web of Science, and arXiv identified 847 records; after deduplication (264 removed) and screening, 47 studies met inclusion criteria (2022–2026, FL/EHDS focus, peer-reviewed or recognized institutional origin). Additional records from institutional websites (n=15), organisations (n=8), and citation searching (n=2) were assessed but did not contribute to the final inclusion set. Adapted from Page et al. (BMJ 2021;372:n71), licensed under CC BY 4.0.

enumerated purposes (scientific research, public health surveillance, AI training for health). The permit validation module is invoked at the beginning of every FL round—not just at training start—to guarantee continuous compliance even if a permit is revoked mid-study or its temporal validity expires. Each validation event is logged as a GDPR Article 30 processing record, creating an immutable audit trail that regulators can inspect.

EHDS Governance Role: This algorithm is the enforcement point between the Governance Layer (Layer 1) and the FL Orchestration Layer (Layer 2). Without it, a permit expiring at round 15 of a 20-round study would allow unauthorized data processing for rounds 16–20.

Failure modes: Each exception type triggers a different response: `PermitExpiredError` terminates the entire study; `PurposeMismatchError` indicates a configuration error requiring researcher intervention; `UnauthorizedCategoryError` may allow continued training on the authorized subset of categories. All failure

events are logged for regulatory audit. Once a round passes permit validation, the aggregation server collects client gradients using the secure protocol described next.

D. Secure Aggregation Protocol

Even though FL prevents raw data sharing, the gradient updates themselves can leak patient information: Zhu et al. [21] demonstrate that gradients can be inverted to reconstruct training images. In the EHDS context, where gradients encode patterns from sensitive health records across 27 Member States, this is an unacceptable privacy risk. Secure aggregation addresses this by ensuring the SPE aggregation server can compute $\sum_k \Delta_k$ without ever observing any individual hospital’s gradient Δ_k .

Algorithm S4 implements this using Shamir’s secret sharing and pairwise masking, ensuring the server observes only the aggregate gradient.

Protocol Phases: (1) Each client splits gradients into K shares using (t, K) -threshold Shamir secret sharing; (2) Clients add pairwise random masks negotiated via ECDH key

Algorithm S1: FL-EHDS FedAvg Training**Input:** Hospitals $\mathcal{H} = \{h_1, \dots, h_K\}$, permit P , rounds T **Output:** Global model $\theta^{(T)}$ **Server executes:**

```

Initialize  $\theta^{(0)}$ 
for round  $t = 1$  to  $T$  do
    // Governance check (Layer 1)
    if not ValidatePermit( $P$ ,  $t$ ) then abort
     $\mathcal{H}_t \leftarrow \text{SelectParticipants}(\mathcal{H})$ 
    for each hospital  $h \in \mathcal{H}_t$  in parallel do
         $\Delta_h^{(t)}, n_h \leftarrow \text{LocalTrain}(h, \theta^{(t-1)})$ 
    // Aggregation with privacy (Layer 2)
     $\theta^{(t)} \leftarrow \theta^{(t-1)} + \frac{1}{\sum_{h \in \mathcal{H}_t} n_h} \sum_{h \in \mathcal{H}_t} n_h \cdot \Delta_h^{(t)}$ 
    LogCompliance( $t, \mathcal{H}_t$ )
return  $\theta^{(T)}$ 

```

LocalTrain(h, θ) at hospital h :

```

// Opt-out filtering (Article 71)
 $\mathcal{D}_h \leftarrow \text{FilterOptedOut}(\mathcal{D}_h, \text{OptOutRegistry})$ 
 $\theta_h \leftarrow \theta$ 
for epoch  $e = 1$  to  $E$  do
    for batch  $\mathcal{B} \in \mathcal{D}_h$  do
         $\theta_h \leftarrow \theta_h - \eta \nabla \mathcal{L}(\theta_h; \mathcal{B})$ 
 $\Delta_h \leftarrow \theta_h - \theta$ 
// Privacy protection (Layer 3)
 $\Delta_h \leftarrow \text{ClipGradient}(\Delta_h, C)$ 
return  $\Delta_h, |\mathcal{D}_h|$ 

```

Algorithm S2: Gaussian DP Mechanism**Input:** Gradient Δ , sensitivity C , privacy budget ϵ, δ **Output:** Noisy gradient $\tilde{\Delta}$

```

// Compute noise scale from Gaussian mechanism
 $\sigma \leftarrow C \cdot \sqrt{2 \ln(1.25/\delta)}/\epsilon$ 
// Add calibrated Gaussian noise to each parameter
for each parameter  $w \in \Delta$  do
     $\tilde{w} \leftarrow w + \mathcal{N}(0, \sigma^2)$ 
// Track cumulative privacy expenditure
PrivacyAccountant.spend( $\epsilon$ )
if PrivacyAccountant.budget_exhausted() then
    raise PrivacyBudgetExhaustedError
return  $\tilde{\Delta}$ 

```

Algorithm S3: Data Permit Validation**Input:** Permit P , round t , requested categories \mathcal{C} **Output:** Boolean validity

```

// Check temporal validity
if CurrentTime() > P.valid_until then
    raise PermitExpiredError
// Check purpose alignment (Article 53)
if P.purpose  $\notin$  AllowedPurposes then
    raise PurposeMismatchError
// Check data category authorization
for each category  $c \in \mathcal{C}$  do
    if  $c \notin P.authorized\_categories$  then
        raise UnauthorizedCategoryError
// Log access for GDPR Article 30
AuditTrail.log(permit= $P$ , round= $t$ , categories= $\mathcal{C}$ )
return True

```

exchange; (3) The server computes the sum—masks cancel out and only the true aggregate remains.

Algorithm S4: Secure Aggregation (Pairwise Masking)**Input:** Client gradients $\{\Delta_1, \dots, \Delta_K\}$, threshold t **Output:** Aggregated gradient Δ_{agg}

```

// Phase 1: ECDH key exchange + Shamir sharing
for each client  $k$  do
    shares $_k \leftarrow \text{ShamirShare}(\Delta_k, t, K)$ 
    Distribute shares $_k$  to other clients
// Phase 2: Add pairwise random masks
for each client  $k$  do
     $\Delta_k \leftarrow \Delta_k + \sum_{j < k} r_{jk} - \sum_{j > k} r_{kj}$ 
// Phase 3: Server reconstructs aggregate
 $\Delta_{agg} \leftarrow \sum_{k=1}^K \Delta_k$ 
// Masks cancel:  $\sum_k \sum_{j < k} r_{jk} - \sum_k \sum_{j > k} r_{kj} = 0$ 
if ActiveClients <  $t$  then
    raise SecureAggregationError
return  $\Delta_{agg}$ 

```

Defense-in-depth: Secure aggregation (Algorithm S4) combined with differential privacy (Algorithm S2) provides layered protection: even if the aggregation server is compromised, it learns only the noisy aggregate—never individual hospital contributions. The combination addresses the unresolved GDPR question of whether model gradients constitute “personal data”: with both mechanisms active, the information available to any single party is provably bounded. The following algorithms address how local training handles EHDS-specific data challenges.

E. FedProx for Non-IID Data

Algorithm S5 extends FedAvg with a proximal term that penalizes local model divergence from the global model [14]. In EHDS cross-border federations, data heterogeneity is a structural feature, not an exception: hospitals in different Member States serve distinct demographics, follow national clinical guidelines, and use different diagnostic thresholds. For instance, heart disease prevalence ranges from 39.2% in Rome to 62.6% in Amsterdam in our experimental setting. Without drift control, local models can diverge so far from the global consensus that aggregation produces a deteriorated global model. The proximal term $\frac{\mu}{2} \|\theta_h - \theta\|^2$ acts as a regularizer that keeps each hospital’s local update within a controlled distance of the global model, balancing personalization with collaboration.

When to use in EHDS: Recommended for federations with moderate non-IID conditions and when client dropout is expected (hospitals may temporarily disconnect). FedProx tolerates partial participation better than FedAvg because the proximal term stabilizes local updates even with fewer training epochs.

Parameter Selection: $\mu = 0$ reduces to FedAvg; $\mu \in [0.01, 0.1]$ provides stable convergence; $\mu > 1$ may prevent local adaptation. The choice of μ should be documented in the data permit application so that the HDAB can assess the expected privacy-utility trade-off.

Algorithm S5: FedProx Local Update

Input: Local data \mathcal{D}_h , global model θ , proximal weight μ
Output: Local update Δ_h

```

 $\theta_h \leftarrow \theta$ 
for epoch  $e = 1$  to  $E$  do
  for batch  $\mathcal{B} \in \mathcal{D}_h$  do
     $g \leftarrow \nabla \mathcal{L}(\theta_h; \mathcal{B})$ 
    // Proximal term:  $\nabla \frac{\mu}{2} \|\theta_h - \theta\|^2$ 
     $g \leftarrow g + \mu(\theta_h - \theta)$ 
     $\theta_h \leftarrow \theta_h - \eta \cdot g$ 
 $\Delta_h \leftarrow \theta_h - \theta$ 
return  $\Delta_h$ 

```

Before any local training begins (whether with FedAvg, FedProx, or any other algorithm), the framework must enforce citizen opt-out rights. The following algorithm ensures this compliance.

F. Article 71 Opt-Out Registry Protocol

Algorithm S6 implements the citizen opt-out mechanism mandated by EHDS Article 71. This article grants every EU citizen the right to object to secondary use of their electronic health data—a fundamental right that must be enforced *before* any gradient computation occurs. The algorithm queries the national opt-out registry maintained by each Member State and removes matching records from the local training dataset.

Granularity levels: (1) *Blanket opt-out*—citizen refuses all secondary use; (2) *Purpose-specific*—e.g., permitting scientific research but blocking commercial analytics; (3) *Category-specific*—e.g., allowing demographics but blocking genomic data. This granularity reflects the EHDS principle that citizens should have meaningful control, not merely a binary yes/no choice.

EHDS Governance Role: Opt-out filtering operates at Layer 3 (Data Holders) before local training. Registry lookups use LRU caching with configurable TTL to minimize latency (<10ms per round) while ensuring timely propagation of new opt-out decisions. All filtering statistics are logged for GDPR Article 30 audit compliance.

Impact on model quality: High opt-out rates reduce training data volume, potentially degrading model performance—particularly for underrepresented subpopulations. The audit log captures filtering statistics to quantify this impact and support transparency reporting. Once opted-out records are excluded, the remaining data must be harmonized into a consistent format before local model training can proceed.

G. FHIR R4 Preprocessing Pipeline

Algorithm S7 standardizes heterogeneous EHR data into FHIR R4 format for ML consumption. This preprocessing step is critical in the EHDS context because only 34% of European healthcare providers currently achieve full FHIR R4 compliance [7]. The remaining 66% use legacy formats (HL7v2, CDA, proprietary CSV exports) that must be harmonized before FL training can proceed on a consistent feature space.

Algorithm S6: Article 71 Opt-Out Filtering

Input: Local dataset \mathcal{D}_h , purpose p , categories \mathcal{C}
Output: Filtered dataset \mathcal{D}'_h

```

// Synchronize with national opt-out registry
OptOutRecords  $\leftarrow$  FetchOptOutRegistry(MemberState)
 $\mathcal{D}'_h \leftarrow \emptyset$ 
for each record  $r \in \mathcal{D}_h$  do
  citizen_id  $\leftarrow$  r.pseudonymized_id
  opted_out  $\leftarrow$  False
  // Check purpose-specific opt-out
  if (citizen_id,  $p$ )  $\in$  OptOutRecords then
    opted_out  $\leftarrow$  True
  // Check category-specific opt-out
  for each  $c \in \mathcal{C}$  do
    if (citizen_id,  $c$ )  $\in$  OptOutRecords then
      opted_out  $\leftarrow$  True
  if not opted_out then
     $\mathcal{D}'_h \leftarrow \mathcal{D}'_h \cup \{r\}$ 
// Log filtering statistics for audit
AuditLog.record(total= $|\mathcal{D}_h|$ , filtered= $|\mathcal{D}'_h|$ )
return  $\mathcal{D}'_h$ 

```

Four-stage pipeline: (1) *Format detection* automatically identifies the source format; (2) *Terminology mapping* converts local codes to international standards (ICD-10 for diagnoses, ATC for medications, LOINC for laboratory results); (3) *FHIR transformation* produces validated FHIR R4 bundles using the six Article 33 data categories (Patient Summary, E-Prescription, Laboratory Results, Medical Imaging, Hospital Discharge, Rare Disease); (4) *Tensor extraction* converts structured FHIR resources into numerical tensors ready for model training.

EHDS Relevance: Without this harmonization step, hospitals in different Member States would produce incompatible feature spaces, making federated aggregation meaningless. The pipeline ensures that a gradient computed in a Finnish hospital is semantically compatible with one from an Italian hospital.

Algorithm S7: FHIR R4 Preprocessing

Input: Raw EHR records \mathcal{R} , feature specification \mathcal{F}
Output: Training tensors (X, y)

```

format  $\leftarrow$  DetectFormat( $\mathcal{R}$ ) // HL7v2, CDA, CSV
parser  $\leftarrow$  GetParser(format)
records  $\leftarrow$  parser.parse( $\mathcal{R}$ )
// Map to standard terminologies
for each  $r \in$  records do
   $r$ .diagnoses  $\leftarrow$  MapToICD10( $r$ .diagnoses)
   $r$ .medications  $\leftarrow$  MapToATC( $r$ .medications)
   $r$ .labs  $\leftarrow$  MapToLOINC( $r$ .labs)
fhir_bundle  $\leftarrow$  ToFHIR(records)
ValidateFHIR(fhir_bundle)
 $X \leftarrow$  ExtractFeatures(fhir_bundle,  $\mathcal{F}$ )
 $X \leftarrow$  StandardScaler.fit_transform( $X$ )
 $y \leftarrow$  ExtractLabels(fhir_bundle)
return  $(X, y)$ 

```

Validation requirements: The FHIR validation step rejects records with missing mandatory fields or invalid terminology codes, ensuring data quality before model training. Rejected

records are logged (without patient-identifiable content) for audit purposes. With harmonized data ready for training, the final core component manages the overall privacy budget across the entire study.

H. Privacy Budget Accountant

Algorithm S8 tracks cumulative privacy expenditure across FL rounds using Rényi Differential Privacy (RDP) moment accounting [26]. In the EHDS governance model, the total privacy budget ε_{total} is a parameter of the data permit: the researcher specifies the desired budget in the permit application, and the HDAB evaluates whether the proposed budget provides sufficient privacy protection for the requested data categories and population size.

Why RDP accounting: Naive DP composition (adding ε per round) yields loose bounds: 20 rounds at $\varepsilon=0.5$ each would consume $\varepsilon=10$ total. RDP provides 5–6 \times tighter bounds [26], [27], meaning the same 20 rounds can achieve the same privacy guarantee with significantly less noise—and therefore better model utility.

Hard budget enforcement: When the cumulative expenditure approaches ε_{total} , the accountant raises a `BudgetExhaustedError` that terminates training. This prevents “privacy bankruptcy”—a situation where continued training would violate the privacy guarantee approved in the data permit. The per-round allocation strategy distributes remaining budget uniformly across remaining rounds, adapting dynamically if training converges faster than expected.

Algorithm S8: RDP Privacy Budget Accountant

Input: Total budget $(\varepsilon_{total}, \delta_{total})$, rounds T

Output: Per-round budget allocation

$\lambda \leftarrow [0] \times \text{MAX_ORDER}$ // Rényi moments
 $\text{rounds_completed} \leftarrow 0$

function AllocateRound():

$\varepsilon_{spent} \leftarrow \text{ComputeEpsilon}(\lambda, \delta_{total})$

$\varepsilon_{remaining} \leftarrow \varepsilon_{total} - \varepsilon_{spent}$

if $\varepsilon_{remaining} < \varepsilon_{min}$ **then**
 raise BudgetExhaustedError

$\varepsilon_t \leftarrow \varepsilon_{remaining} / (T - \text{rounds_completed})$

return ε_t

function RecordRound(σ, q):

for $\text{order} = 1$ to MAX_ORDER **do**

$\lambda[\text{order}] += \text{ComputeMoment}(\text{order}, \sigma, q)$

$\text{rounds_completed} += 1$

III. SUPPLEMENTARY EXPERIMENTAL FIGURES

This section presents detailed experimental results from the FL-EHDS benchmark suite. All figures are generated from real experimental runs available in the repository.

Note on experimental configurations: Figures 2–9 were generated from an extended 50-round, 5-client training run using the framework’s synthetic EHDS scenario (simulated European hospitals: Rome, Amsterdam, Berlin, Madrid, Paris). These complement the main paper’s 20-round experiments on Heart Disease UCI (4 real hospitals) and Diabetes (5 Dirichlet-partitioned clients). The 50-round configuration illustrates

longer-horizon convergence properties, client participation dynamics, and gradient evolution patterns that are not visible in the shorter 20-round evaluation.

A. Hospital Data Distribution

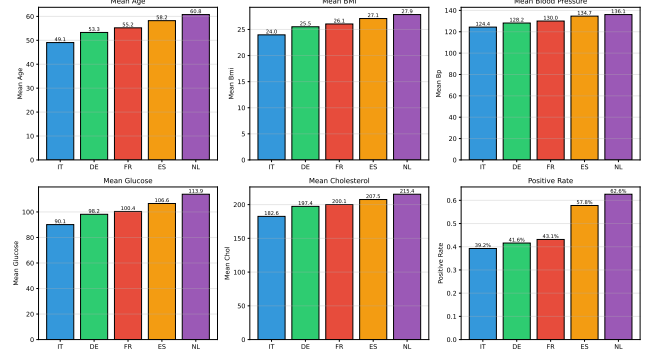


Fig. 2. Data distribution across hospitals. Notable heterogeneity: Amsterdam shows older population (60.8 years mean age) with higher positive rate (62.6%) compared to Rome (49.1 years, 39.2%). This reflects realistic cross-border EHDS variability.

B. Per-Client Training Time

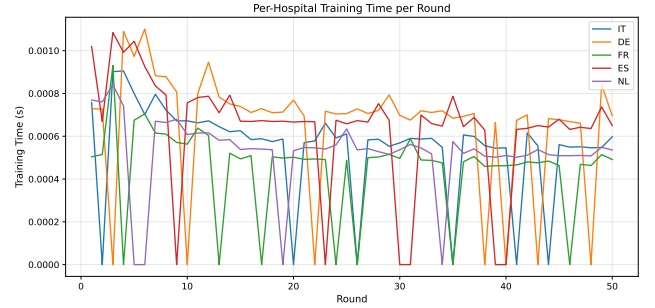


Fig. 3. Per-client training time per round. Larger hospitals (Berlin: 500 samples) exhibit slightly longer training times. The adaptive training engine compensates by adjusting batch sizes for stragglers.

C. Client Participation Matrix

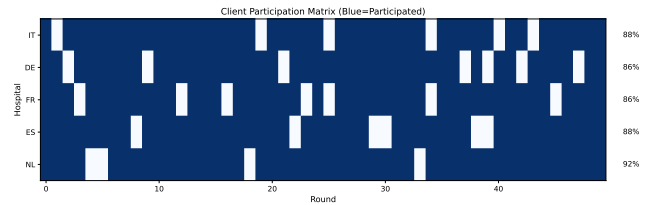


Fig. 4. Client participation matrix (50 rounds \times 5 clients). Participation rates: IT 88%, DE 86%, FR 86%, ES 88%, NL 92%. The framework tolerates 10–15% dropout per round while maintaining convergence.

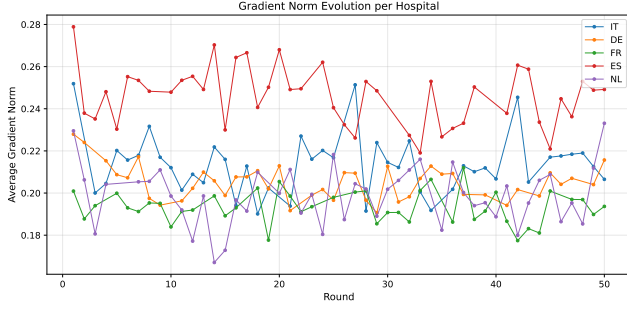


Fig. 5. Gradient norm evolution per client over 50 rounds. All clients show decreasing trends indicating stable convergence. Clipping threshold $C=1.0$ bounds extreme values for DP compatibility.

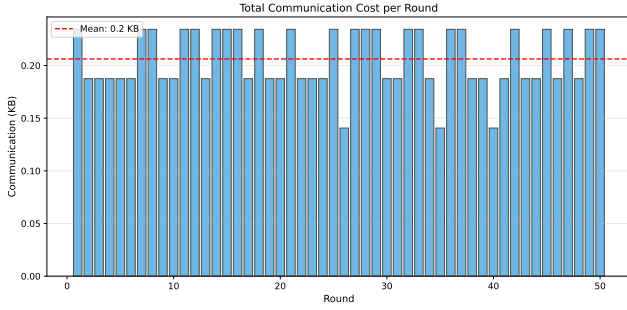


Fig. 6. Cumulative communication cost per round. Linear scaling with participating clients (3.5 KB/client/round). Total 50-round overhead: 875 KB for 5 clients—feasible even for bandwidth-constrained environments.

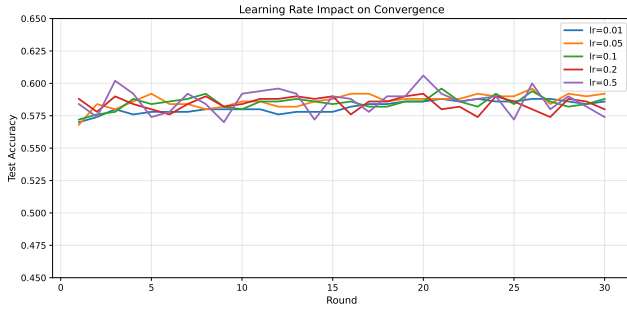


Fig. 7. Learning rate sensitivity analysis. $\eta=0.01$: slow convergence (53.8% at round 50). $\eta=0.1$: optimal (58.6%). $\eta=0.5$: instability with oscillations.

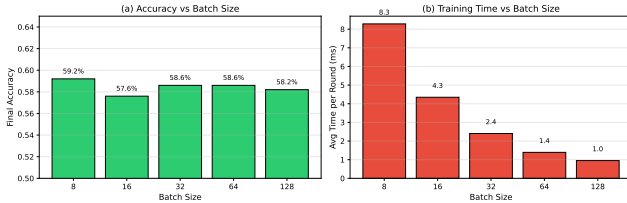


Fig. 8. Batch size impact on convergence. Smaller batches (8–16) provide noisier gradients but faster initial progress. Batch size 32 balances gradient quality and computational efficiency.

D. Gradient Norm Evolution

E. Communication Cost Analysis

F. Learning Rate Sensitivity

G. Batch Size Impact

H. Per-Client Accuracy Trajectories

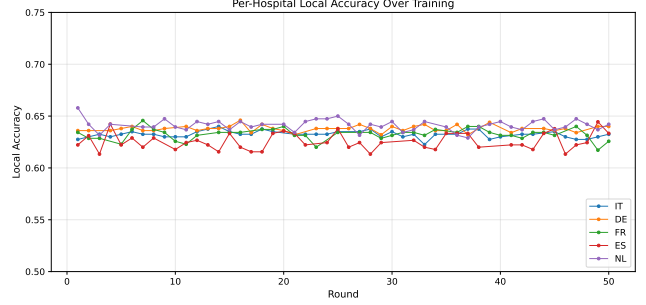


Fig. 9. Per-client accuracy over training rounds. Variance reflects non-IID data: NL (older, higher-risk population) reaches 64% accuracy while FR (mid-range demographics) stabilizes at 55%.

IV. DATASET LANDSCAPE

The FL-EHDS framework supports 19 healthcare datasets spanning four modalities. Table I provides a comprehensive overview. This diversity enables evaluation across multiple EHDS-relevant dimensions: data scale (120–253K samples), feature dimensionality (9–30 tabular, high-dimensional imaging), task complexity (binary to 5-class), partition strategies (natural hospital-based and synthetic Dirichlet), and interoperability standards (CSV, FHIR R4, OMOP-CDM). Experimentally evaluated datasets in the main paper are marked with \checkmark .

V. EXTENDED ALGORITHM COMPARISON

A. Algorithms Evaluated

We compare foundational FL algorithms plus 2022–2025 advances:

Foundational: FedAvg [13], FedProx [14], SCAFFOLD [31], FedAdam/FedYogi/FedAdagrad [33].

Recent (2022–2025): FedLC [37] (logit calibration for label skew), FedSAM [36] (flat minima), FedDecorr [38] (decorrelation against dimensional collapse), FedSpeed [39] (fewer rounds), FedExp [40] (server-side acceleration), FedLE-SAM [41] (globally-guided SAM, ICML 2024 Spotlight), HPFL [42] (personalized classifiers, ICLR 2025).

B. Non-IID Configuration

Data heterogeneity is controlled via Dirichlet distribution with α :

- $\alpha = 0.1$: **Extreme non-IID**—highly skewed label distributions
- $\alpha = 0.5$: **High non-IID**—significant heterogeneity
- $\alpha = 1.0$: **Moderate non-IID**—balanced heterogeneity
- $\alpha = 10.0$: **Near-IID**—approximately uniform

TABLE I
FL-EHDS DATASET LANDSCAPE: COMPLETE FRAMEWORK COVERAGE

Dataset	Type	Samples	Feat.	Classes	FL Partition	EHDS Category (Art. 33)	EHDS Level	Eval.
<i>A. Tabular — Clinical EHR</i>								
Diabetes 130-US	Tabular	101,766	22	2	Dirichlet ($\alpha=0.5$)	EHR, ICD-9, medications	L2: FHIR-mappable	✓
Heart Disease UCI	Tabular	920	13	2	Natural (4 hospitals)	Vitals, ECG, lab results	L2: FHIR-mappable	✓
PTB-XL ECG [†]	Tabular	21,799	9	5	Natural (52 EU sites)	SCP-ECG (EN 1064), diagnostics	L2: FHIR-mappable	✓
Cardiovascular Disease	Tabular	70,000	11	2	Dirichlet ($\alpha=0.5$)	Vitals, lab, risk factors	L2: FHIR-mappable	✓
Breast Cancer Wisconsin	Tabular	569	30	2	Dirichlet ($\alpha=0.5$)	Pathology (FNA cytology)	L2: FHIR-mappable	✓
Stroke Prediction	Tabular	5,110	10	2	Dirichlet	Cardiovascular risk factors	L2: FHIR-mappable	—
CDC Diabetes BRFSS	Tabular	253,680	21	2	Dirichlet	Population health survey	L2: FHIR-mappable	—
CKD UCI	Tabular	400	24	2	Dirichlet	Renal panel, comorbidities	L2: FHIR-mappable	—
Cirrhosis Mayo	Tabular	418	18	2	Dirichlet	Hepatology, drug trial	L2: FHIR-mappable	—
<i>B. Tabular — FHIR-Native</i>								
Synthea FHIR R4	FHIR	1,180	14	2	Hospital profile	Patient, Condition, Encounter	L1: FHIR-native	qual.
SMART Bulk FHIR	FHIR	120	12	2	Single export	NDJSON Bulk Data (Art. 46)	L1: FHIR-native	qual.
<i>C. Generated Pipelines (in-memory)</i>								
FHIR R4 Synthetic	Gen.	config.	10	2	Hospital profile	Generated FHIR bundles	L1: FHIR-native	qual.
OMOP-CDM Harmonized	Gen.	config.	~36	2	Cross-border	Vocabulary harmonization	L3: OMOP	qual.
<i>D. Medical Imaging</i>								
Chest X-ray	Imaging	5,856	—	2	Dirichlet ($\alpha=0.5$)	Radiology (DICOM)	L4: Imaging	✓
Brain Tumor MRI	Imaging	3,064	—	4	Dirichlet ($\alpha=0.5$)	Neuro-imaging (DICOM)	L4: Imaging	✓
Skin Cancer	Imaging	3,297	—	2	Dirichlet ($\alpha=0.5$)	Dermatology (DICOM)	L4: Imaging	✓
Diabetic Retinopathy	Imaging	35,126	—	5	Dirichlet	Ophthalmology (DICOM)	L4: Imaging	—
Brain Tumor MRI (alt.)	Imaging	3,264	—	4	Dirichlet	Neuro-imaging (DICOM)	L4: Imaging	—
ISIC Skin Lesions	Imaging	2,357	—	9	Dirichlet	Dermatology (DICOM)	L4: Imaging	—

EHDS Levels: L1 = FHIR-native (Art. 46 compliant); L2 = FHIR-mappable (standard clinical features with FHIR mapping in metadata); L3 = OMOP-CDM harmonized (cross-border vocabulary alignment, Art. 50); L4 = Medical imaging (DICOM, Art. 33 “medical images”).

[†]PTB-XL: European-origin dataset (PTB, Berlin, Germany) with SCP-ECG coding (EN 1064). 52 recording sites enable natural hospital-based FL partitioning—the strongest EHDS benchmark in the framework.

Eval.: ✓ = quantitative experimental evaluation (P1.2); qual. = qualitative pipeline validation; — = supported but not evaluated in current paper. **config.** = configurable sample count.

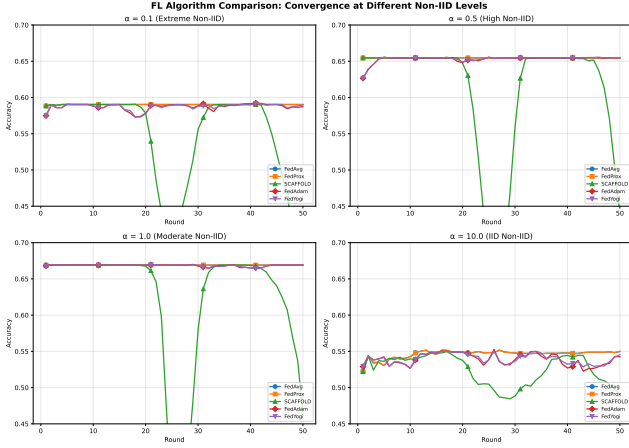


Fig. 10. Algorithm convergence across non-IID levels ($\alpha \in \{0.1, 0.5, 1.0, 10.0\}$). SCAFFOLD and adaptive methods show superior stability under extreme heterogeneity.

C. Convergence at Different Heterogeneity Levels

Findings: (1) At $\alpha=0.1$, SCAFFOLD achieves most stable convergence via variance reduction. (2) FedProx provides marginal improvement over FedAvg at $\alpha=0.5-1.0$. (3) Adaptive methods (FedAdam, FedYogi) excel in near-IID but may oscillate under extreme heterogeneity. (4) FedAvg remains competitive in near-IID, suitable for homogeneous federations.

D. Final Accuracy vs. Heterogeneity

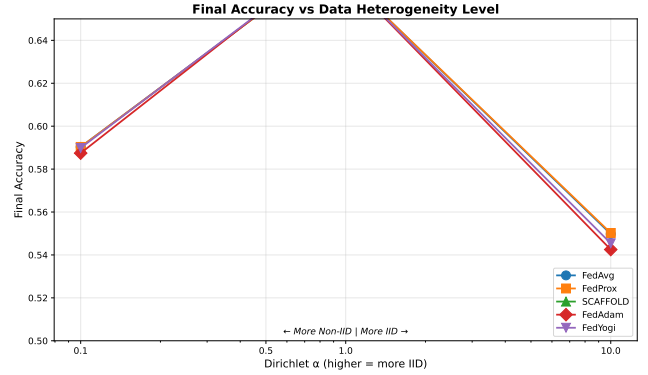


Fig. 11. Final accuracy vs. Dirichlet α . All algorithms degrade under extreme non-IID. SCAFFOLD shows smallest gap between $\alpha=0.1$ and $\alpha=10$.

E. Convergence Speed

F. Algorithm Selection Guidelines

Table II maps EHDS deployment scenarios to recommended algorithms.

VI. ADVANCED FL PARADIGMS

The core FL-EHDS pipeline (Section II) addresses the standard “horizontal” FL scenario where all hospitals share

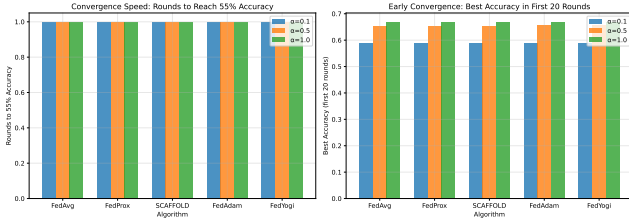


Fig. 12. Convergence speed comparison. Left: rounds to 55% accuracy. Right: best accuracy in first 20 rounds. Adaptive methods converge faster but may plateau.

TABLE II
ALGORITHM SELECTION FOR EHDS DEPLOYMENTS

EHDS Scenario	Algorithm	Rationale
Homogeneous MS	FedAvg	Simplicity, proven
Heterogeneous MS	SCAFFOLD	Variance reduction
Resource-limited	FedAdam	Fast convergence
Privacy-critical	FedAvg + DP	Well-studied bounds
Sparse participation	FedProx	Dropout resilience
Label-imbalanced	FedLC	Class-freq. calib.
Deep models, non-IID	FedDecorr	Dim. collapse prev.
Comm.-constrained	FedSpeed	Fewer rounds
No client changes	FedExP	Server-side only
SAM + global drift	FedLESAM	Global flatness
Per-hosp. classif.	HPFL	Local boundaries

MS = Member States. Scenarios may combine: heterogeneous + privacy-critical \rightarrow SCAFFOLD + DP.

the same feature schema. However, real EHDS deployments will encounter more complex configurations: institutions with complementary features for the same patients (vertical FL), adversarial participants (Byzantine resilience), evolving data distributions over the 2025–2031 timeline (continual FL), heterogeneous clinical objectives (multi-task FL), and the hierarchical governance structure of the EU itself (hierarchical FL). This section presents the advanced paradigms implemented in the reference framework, each motivated by a specific EHDS deployment challenge. Algorithms S9–S13 formalize the core mechanisms.

A. Vertical Federated Learning

Vertical FL addresses scenarios where institutions hold *different features* for the *same patients*—a common situation in EHDS cross-border analytics. For example, a hospital may hold demographics and diagnoses, a laboratory holds test results, and a pharmacy holds prescription histories. Under EHDS Article 33, these correspond to different data categories (Patient Summary, Laboratory Results, E-Prescription) that may be held by different data holders within the same or different Member States.

Private Set Intersection (PSI): Before training, the participating institutions must identify their common patients without revealing their full patient lists. RSA-based PSI achieves this with $O(n \log n)$ complexity using pseudonymized identifiers,

ensuring EHDS compliance: no institution learns which patients the other holds beyond the intersection.

Split Learning: Algorithm S9 implements the forward pass in split learning, where each party computes activations on its local features up to a “cut layer,” then the server concatenates activations to produce the final prediction. Only intermediate representations (not raw data) cross institutional boundaries.

Algorithm S9: Split Learning Forward Pass

Input: Features X_A, X_B at parties A, B; cut layer k

Output: Prediction \hat{y}

```

 $h_A \leftarrow f_{1:k}^A(X_A)$  // Party A: features  $\rightarrow$  activations
 $h_B \leftarrow f_{1:k}^B(X_B)$  // Party B: features  $\rightarrow$  activations
 $h \leftarrow \text{Concat}(h_A, h_B)$ 
 $\hat{y} \leftarrow f_{k+1:L}(h)$  // Server: cut layer  $\rightarrow$  output
return  $\hat{y}$ 

```

B. Byzantine-Resilient Aggregation

In a cross-border EHDS federation spanning 27 Member States, the aggregation server cannot blindly trust every participant. A compromised institution—whether through malicious intent, software bugs, or data corruption—could submit adversarial gradient updates that poison the global model, potentially affecting clinical decisions across the entire federation. Byzantine-resilient aggregation protects model integrity by detecting and excluding anomalous updates.

Algorithm S10 implements Krum, which selects the gradient closest to $n - f - 2$ nearest neighbors, effectively filtering outliers. Six defense methods protect against up to $f < n/3$ adversarial clients:

Algorithm S10: Krum Byzantine Defense

Input: Gradients $\{g_1, \dots, g_n\}$, Byzantine bound f

Output: Selected gradient g^*

```

for each gradient  $g_i$  do
   $D_i \leftarrow \{\|g_i - g_j\|^2 : j \neq i\}$ 
   $s_i \leftarrow \sum_{d \in \text{smallest}_{n-f-2}(D_i)} d$ 
 $g^* \leftarrow g_{\arg \min_i s_i}$ 
return  $g^*$ 

```

Other methods: **Trimmed Mean** (removes β -fraction extreme values per coordinate), **Coordinate-wise Median** (robust estimator), **Bulyan** (two-stage Krum + trimmed mean), **FLTrust** (server-guided trust weighting), **FLAME** (clustering-based). Attack simulation: label flipping, gradient scaling, additive noise, sign flipping, model replacement.

C. Continual Federated Learning

The EHDS is designed for long-term operation (2025–2031 and beyond), during which healthcare data distributions will evolve: new diseases emerge (as demonstrated by COVID-19), clinical protocols change, and demographic compositions shift. A model trained in 2027 may perform poorly on 2029 data if it has “forgotten” how to handle earlier patterns. Continual Federated Learning addresses this *catastrophic forgetting*

problem by preserving knowledge from previous training tasks while adapting to new data.

The Elastic Weight Consolidation (EWC) loss function adds a quadratic penalty:

$$\mathcal{L}_{EWC}(\theta) = \mathcal{L}(\theta) + \frac{\lambda}{2} \sum_i F_i(\theta_i - \theta_i^*)^2$$

where F_i is the Fisher Information for parameter i and θ^* are optimal parameters for previous tasks.

Additional strategies: Learning without Forgetting (LwF), Experience Replay, drift detection (ADWIN, Page-Hinkley) triggering adaptation.

D. Multi-Task Federated Learning

In EHDS cross-border studies, different hospitals may pursue related but distinct clinical objectives from the same data. A cardiology network might simultaneously predict heart failure risk (Hospital A), readmission probability (Hospital B), and medication response (Hospital C). Multi-Task FL enables these institutions to collaborate on shared feature representations while maintaining task-specific prediction heads.

Architectures: **Hard Parameter Sharing** (common feature extractor, task-specific heads), **Soft Parameter Sharing** (separate networks with similarity regularization), **FedMTL** (dynamic task relationship learning).

E. Hierarchical Federated Learning

The EHDS governance structure is inherently hierarchical: individual hospitals report to regional health authorities, which coordinate under national HDABs, which in turn connect to the EU-level HealthData@EU infrastructure. Hierarchical FL mirrors this governance topology, aggregating gradients at intermediate levels before reaching the central server. This reduces communication costs (hospitals communicate with regional aggregators, not directly with the EU server) and aligns FL operations with the jurisdictional boundaries of HDABs.

Four-tier hierarchy reflecting EU governance:

- 1) **Client Tier:** Individual hospitals/data holders
- 2) **Regional Tier:** Regional aggregators (e.g., Lombardy, Bavaria)
- 3) **National Tier:** National HDABs coordinate Member State aggregation
- 4) **EU Tier:** HealthData@EU central aggregator

Benefits: reduced communication costs (hospitals → regional, not directly EU), alignment with EHDS governance where HDABs have national jurisdiction.

F. Personalized Federated Learning

A single global model may underperform at individual hospitals because clinical populations differ substantially across Member States. Personalized FL maintains both a global model (encoding shared medical knowledge) and hospital-specific local models (adapted to local demographics and clinical practices). Algorithm S11 shows pFedMe [35] as a representative personalized method, using Moreau envelopes

to balance personalization with global knowledge: the regularization term $\lambda(\theta_k - \theta)$ pulls the local model toward the global consensus, while local gradient descent adapts to hospital-specific data patterns. Ditto [34] follows a similar dual-model principle but with a simpler formulation: it trains a personalized model regularized toward the global model via $\frac{\lambda}{2}\|\theta_k - \theta\|^2$. In our experiments, Ditto—the best-performing personalized method—achieves 75.1% accuracy on Heart Disease, a 12.6pp improvement over FedAvg, precisely because it learns hospital-specific decision boundaries.

Algorithm S11: pFedMe Local Update

Input: Data \mathcal{D}_k , global θ , personal θ_k , λ , η

Output: Updated personal model θ'_k

```

for  $i = 1$  to  $R$  do
     $\theta_k \leftarrow \theta_k - \eta \nabla \mathcal{L}(\theta_k; \mathcal{D}_k)$ 
    // Moreau envelope: balance with global
     $\theta'_k \leftarrow \theta_k - \lambda(\theta_k - \theta)$ 
     $g_k \leftarrow \lambda(\theta - \theta'_k)$ 
return  $\theta'_k, g_k$ 

```

Other approaches: **FedPer** (shared base, local personalization layers), **Per-FedAvg** (MAML-based meta-learning), **APFL** (adaptive mixing α between global and local), **Ditto** (personalization regularization).

EHDS Relevance: Member States have different healthcare systems, disease prevalence, and clinical practices. Personalized FL enables institution-specific adaptation while benefiting from collaborative training.

G. Asynchronous Federated Learning

Standard synchronous FL requires all participating hospitals to complete local training before the server can aggregate. In an EHDS federation spanning 27 Member States with heterogeneous computational resources, this creates a “straggler” problem: a resource-constrained rural hospital delays the entire federation. Asynchronous FL eliminates this bottleneck by allowing the server to aggregate updates as they arrive, weighting stale updates (from slow clients) less heavily. Algorithm S12 implements polynomial staleness weighting: an update computed τ rounds ago receives weight $(1 + \tau)^{-a}$, ensuring that fresher updates contribute more while still incorporating information from slower participants.

Algorithm S12: FedAsync with Staleness Weighting

Input: Client update Δ_k , client round t_k , server round t

Output: Updated global model θ

```

 $\tau \leftarrow t - t_k$  // Staleness
 $\alpha \leftarrow (1 + \tau)^{-a}$  // Polynomial decay,  $a > 0$ 
 $\theta \leftarrow \theta + \alpha \cdot \eta \cdot \Delta_k$ 
return  $\theta$ 

```

Staleness functions: Constant ($\alpha=1$), Polynomial ($(1+\tau)^{-a}$), Exponential ($e^{-a\tau}$), Hinge (1 if $\tau \leq \tau_{max}$, else 0). Additional: FedBuff (buffered async), semi-async (wait for α -fraction of clients).

H. Fairness-Aware Federated Learning

The EHDS serves 450 million citizens across Member States with different population sizes, disease prevalence, and healthcare quality. Standard FL optimizes average performance, which can disproportionately favor large hospitals with more data while neglecting smaller institutions or underrepresented patient populations. This creates a “digital health equity” concern: a model that achieves 85% accuracy for a large German hospital but only 55% for a small Romanian clinic is not equitable. Algorithm S13 implements q-FedAvg, which reweights client contributions by their loss: hospitals where the model performs poorly receive higher aggregation weights, pulling the global model toward equitable performance across all participants.

Algorithm S13: q-FedAvg Fair Aggregation

Input: Losses $\{L_1, \dots, L_K\}$, updates $\{\Delta_1, \dots, \Delta_K\}$, q
Output: Fair aggregated update Δ

for each client k do
 $w_k \leftarrow L_k^q$ // Higher loss \rightarrow higher weight
 $W \leftarrow \sum_k w_k$; $w_k \leftarrow w_k / W$
 $\Delta \leftarrow \sum_k w_k \cdot \Delta_k$
return Δ

Fairness metrics: Performance Variance ($\text{Var}(\{L_k\})$), Worst-case Loss ($\max_k L_k$), Demographic Parity Gap, Equalized Odds Gap. Additional methods: AFL, FedMGDA+, TERM, FairFed.

VII. INFRASTRUCTURE COMPONENTS

Deploying FL across 27 EU Member States requires production-grade infrastructure: reliable communication channels between hospitals and the SPE aggregator, efficient serialization of gradient tensors, distributed coordination for concurrent studies, and comprehensive monitoring with EHDS-specific alerting. This section describes the infrastructure components implemented in the reference framework, each designed to operate within the constraints of cross-border healthcare networks (firewalls, bandwidth limitations, regulatory requirements).

A. Communication Layer

The communication layer must bridge heterogeneous network environments: high-bandwidth data center connections between national HDABs, moderate hospital-to-aggregator links, and potentially bandwidth-constrained rural clinics. The framework supports two transport protocols selectable per deployment, with configurable compression and retry policies.

gRPC: Bidirectional streaming, Protocol Buffers (30% bandwidth reduction vs. JSON), HTTP/2 multiplexing. Ideal for data center deployments.

WebSocket: Browser-compatible, firewall-friendly (standard HTTP upgrade), event-driven. Ideal for edge deployments and browser-based participation.

Selection criteria: gRPC is recommended for production EHDS deployments where both endpoints support HTTP/2 (typical for hospital-to-national aggregator links). WebSocket

Communication Manager Configuration

```
transport: gRPC | WebSocket
compression: gzip | lz4 | zstd | none
chunk_size: 1MB
retry_policy:
  max_retries: 3
  backoff: exponential
  base_delay: 1s
connection_pool:
  max_connections: 100
  idle_timeout: 300s
```

is preferred when traffic must traverse web application firewalls or when browser-based dashboards participate directly in federation monitoring.

B. Serialization

Binary Format: Tensor metadata + raw binary, 30% smaller than JSON, 15% smaller than pickle, cross-platform (Python, C++, Java).

Delta Serialization: Transmits only changed parameters, sparse encoding, up to 90% bandwidth reduction for fine-tuning.

EHDS-Compliant: Embeds permit ID, timestamp, provenance; cryptographic signatures; GDPR Article 30 audit fields.

C. Caching Layer

In production EHDS deployments, multiple FL studies may run concurrently on overlapping data holders. A distributed locking mechanism prevents race conditions during gradient aggregation—ensuring that two concurrent studies do not interfere with each other’s model updates. Algorithm S14 implements Redis-based distributed locking with TTL-based automatic release, preventing deadlocks if a server node fails mid-aggregation.

Algorithm S14: Distributed Lock for Aggregation

Input: Lock name, TTL, client ID
Output: Lock acquired (boolean)

$\text{acquired} \leftarrow \text{Redis.SET}(\text{lock_name}, \text{client_id}, \text{NX}, \text{EX}=\text{TTL})$
if acquired **then**
 PerformAggregation()
if $\text{Redis.GET}(\text{lock_name}) == \text{client_id}$ **then**
 Redis.DEL(lock_name)
return acquired

Redis-based caching: model checkpoints, client states, real-time metrics. Features: LRU/LFU/TTL eviction, distributed locking, automatic serialization, cache warming.

D. Orchestration

Kubernetes: Deploys FL clients/aggregators as pods, HPA for elastic scaling, ConfigMaps for hyperparameters, Secrets for HDAB API keys.

Ray: Actor-based FL, automatic fault tolerance, Ray Tune for federated HPO, Object Store for gradient sharing.

Auto-Scaling: Reactive (queue depth/latency), Predictive (ML-based forecasting), Scheduled (time-based patterns).

E. Monitoring

Prometheus Metrics: Counters (rounds_total, permits_validated), Gauges (active_clients, privacy_budget_remaining), Histograms (round_duration, communication_latency), Summaries (gradient_norm_quantiles).

Grafana Dashboards: FL training progress, client health, latency heatmaps, privacy budget consumption, EHDS compliance status.

Alerting: Privacy budget exhaustion, client dropout threshold, model divergence, permit expiration.

F. Model Watermarking

IP protection for FL models trained on EHDS data: **Spread Spectrum** (frequency domain, robust to fine-tuning), **LSB** (low-order weight bits), **Backdoor-based** (input-output ownership proof), **Passport Layers** (dedicated ownership encoding).

G. Cross-Silo Enhancements

EHDS federations are inherently cross-silo: each participant is an institution (hospital, registry, research center) with significant computational resources, distinct data distributions, and long-term participation commitments. This differs from cross-device FL (e.g., mobile phones) and enables advanced optimization strategies.

Multi-Model Federation: Weighted voting, stacking, mixture of experts with diversity enforcement.

Automatic Algorithm Selection: The 17 FL algorithms in the framework have different strengths depending on the federation characteristics (heterogeneity level, number of participants, communication budget). Algorithm S15 implements adaptive aggregation selection via multi-armed bandit (UCB/Thompson Sampling), automatically switching algorithms mid-training if performance metrics indicate a better alternative. A cooldown period prevents oscillation between strategies.

Algorithm S15: Adaptive Aggregation

```
Input: Client updates, metrics history, cooldown
Output: Aggregated model, selected algorithm
score ← WeightedScore(loss, accuracy, variance, conv.)
if RoundsSinceSwitch > Cooldown then
  for each candidate ∈ Algorithms do
    alt ← EstimatePerformance(candidate)
    if alt > score + Threshold then
      SwitchTo(candidate)
  aggregated ← CurrentAlgo.Aggregate(updates)
return aggregated
```

VIII. EXTENDED EHDS INTEROPERABILITY

A. OMOP Common Data Model

OMOP CDM v5.4 provides standardized analytical format used by European research networks (EHDEN, OHDSI).

ETL Pipelines: Transform source EHR to OMOP. **Vocabulary Mapping:** SNOMED, ICD10, LOINC, RxNorm. **Cohort Definitions:** ATLAS-compatible SQL generation. **Feature Extraction:** FeatureExtraction package for ML-ready datasets.

FL Integration: (1) Each hospital transforms local EHR to OMOP; (2) Feature extraction produces identical schema; (3) FL training on homogeneous feature spaces.

B. IHE Integration Profiles

ATNA: TLS mutual authentication, syslog audit messages (RFC 5424), maps to GDPR Article 30.

BPPC: Maps Article 71 opt-out to consent documents, XDS.b integration, consent enforcement at FL initiation.

XCA: Cross-border document query/retrieve, Initiating/Responding Gateways, patient identity correlation.

PIX/PDQ: Patient matching across boundaries, pseudonymization-aware identity management, national eHealth integration.

XUA: SAML 2.0 federated authentication, role-based access control, HDAB authorization token propagation.

C. Cross-Border Data Exchange

Message Formats: EHDS Data Permit Exchange Format (JSON-LD), Federated Query Protocol (SPARQL Federation), Model Update Message Format (Protocol Buffers).

Security: eIDAS-compliant electronic signatures, TLS 1.3, certificate-based authentication (EU trust framework).

Metadata: DCAT-AP Health extension, W3C PROV-O provenance, EMA data quality indicators.

D. Interoperability Architecture

Figure 13 presents the complete interoperability architecture, showing how heterogeneous data sources across EU Member States are harmonized through multiple standards layers before reaching the FL training engine. The architecture reflects a key EHDS challenge: real-world healthcare institutions use diverse formats, terminologies, and exchange protocols that must be reconciled to produce a consistent feature space for federated model training.

IX. CLINICAL IMAGING: EXTENDED DETAILS

A. Datasets

Three clinical imaging datasets cover representative EHDS scenarios:

- **Chest X-ray** [48]: 5,860 pediatric radiographs (NORMAL/PNEUMONIA, 2.7:1 imbalance)
- **Brain Tumor MRI:** 3,064 T1-weighted CE MRI slices (3-class: glioma, meningioma, pituitary)
- **Skin Cancer:** 3,297 dermoscopy images (binary benign/malignant)

B. Model Architectures

HealthcareResNet: ResNet-18 [47] pretrained on ImageNet, GroupNorm replacing BatchNorm for FL stability. FedBN [46] skips normalization during aggregation. Partial backbone freeze (level 1). ~11.2M parameters.

HealthcareCNN: 5-block CNN with GroupNorm, progressive channels (32→512), graduated Dropout (0.15→0.3). Classifier: Flatten→FC(512)→FC(128)→FC(K). ~12M parameters.

Data augmentation: random horizontal flip, rotation ($\pm 15^\circ$), brightness jitter ($\pm 10\%$). ImageNet normalization.

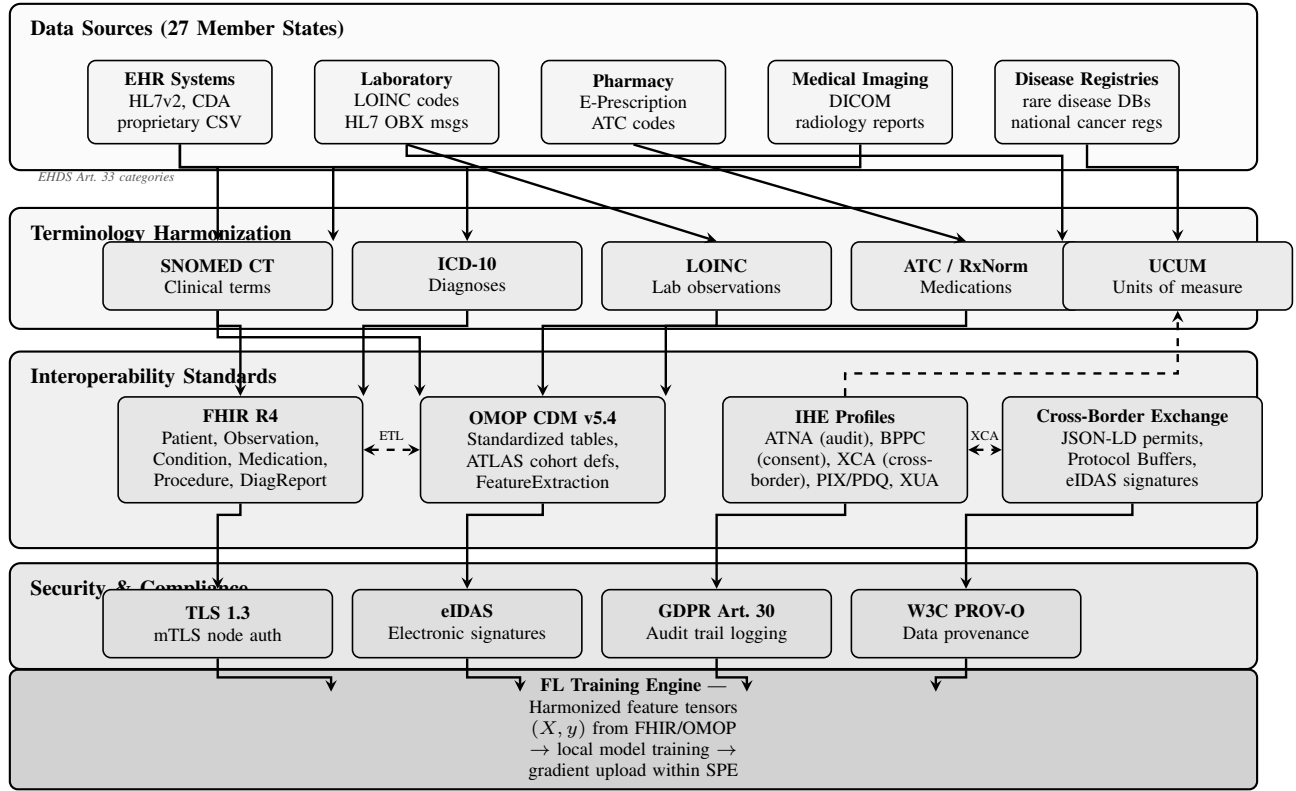


Fig. 13. EHDS interoperability architecture for FL-based secondary use. Data from heterogeneous sources across 27 Member States (top) flows through terminology harmonization (SNOMED CT, ICD-10, LOINC, ATC, UCUM), then through interoperability standards (FHIR R4 for structured data exchange, OMOP CDM for observational research, IHE profiles for cross-institutional workflows, cross-border exchange protocols with eIDAS signatures). A security and compliance layer enforces TLS 1.3 mutual authentication, eIDAS electronic signatures for permits, GDPR Article 30 audit logging, and W3C PROV-O data provenance before harmonized feature tensors reach the FL training engine. Bidirectional ETL between FHIR and OMOP enables institutions to use either standard based on their existing infrastructure.

C. V2 Experimental Configuration

- 5 hospitals, 25 rounds, 3 local epochs, batch size 32
- Adam optimizer (lr=0.001), early stopping (patience=6)
- Non-IID via Dirichlet $\alpha=0.5$
- FedBN enabled, partial backbone freeze (level 1)
- 7 algorithms: FedAvg, FedProx, Ditto, FedLC, FedExP, FedLESAM, HPFL
- 3 seeds per configuration (42, 123, 456)
- Total: $7 \times 5 \text{ datasets} \times 3 \text{ seeds} = 105 \text{ experiments}$

D. Reproducibility

All experiments are fully reproducible:

```
cd fl-ehds-framework
# Full experiments (7 algo x 5 datasets x 3 seeds)
python -m benchmarks.run_full_experiments
# Quick validation (~1-2h)
python -m benchmarks.run_full_experiments --quick
# Resume after interruption
python -m benchmarks.run_full_experiments --resume
```

Results, checkpoints, and logs are auto-saved to `benchmarks/paper_results/`.

Repository: <https://github.com/FabioLiberti/FL-EHDS-FLICS2026>

X. DETAILED ARCHITECTURE DESCRIPTION

This section provides a comprehensive technical specification of the FL-EHDS three-layer architecture, detailing all modules, services, protocols, and standards implemented in the reference framework. Figure ?? in the main paper provides the high-level view; Figure 14 presents the detailed component-level diagram; the description below enumerates every component with its specific technical parameters.

A. Layer 1: Governance

Four principal modules comprise the governance layer:

1.1 HDAB Integration (per Member State). Each Health Data Access Body instance implements: OAuth2/mTLS authentication with bearer token management (refresh tokens, scopes: `permits:read`, `permits:write`); a permit store with SQLite persistence backend; configurable HDAB strictness level (scale 1–5, where DE=5, ES=2); and national privacy constraints including per-jurisdiction differential privacy bounds (ϵ_{\max} : DE=1.0, FR=3.0, IT=5.0).

1.2 Data Permit Manager (Article 53). Manages the complete permit lifecycle: PENDING → APPROVED → ACTIVE → EXPIRED/REVOKED. Validates permitted purposes (scientific research, public health surveillance, AI system development, personalized medicine, official statistics) against au-

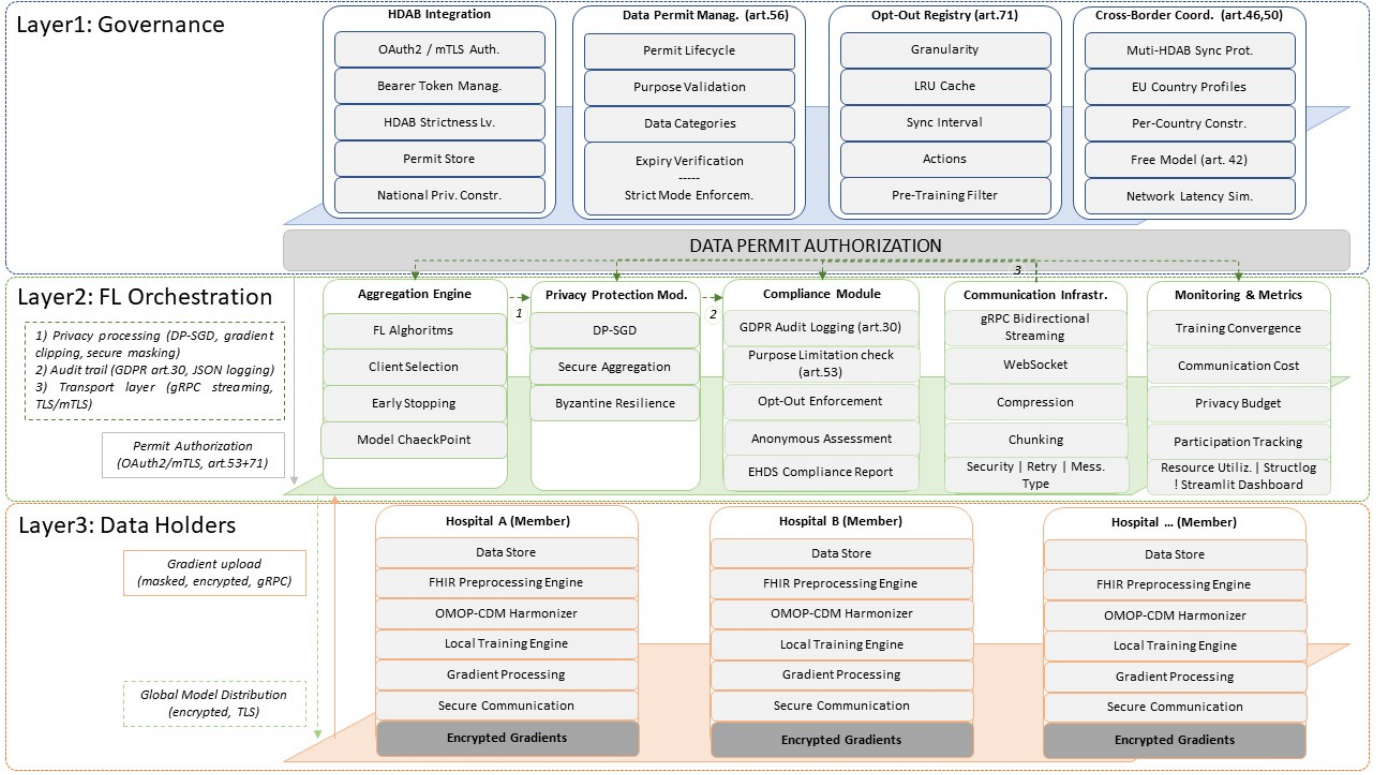


Fig. 14. FL-EHDS detailed architecture with component-level specifications. Layer 1 (Governance) includes four modules: HDAB Integration with OAuth2/mTLS authentication, Data Permit Manager (Article 53) with lifecycle management, Opt-Out Registry (Article 71) with LRU-cached filtering, and Cross-Border Coordinator (Articles 46, 50) with per-country privacy constraints. Layer 2 (FL Orchestration) operates within the Secure Processing Environment (SPE) with five modules: Aggregation Engine (17 FL algorithms), Privacy Protection (DP-SGD, Secure Aggregation, Byzantine Resilience), Compliance Module (GDPR audit logging), Communication Infrastructure (gRPC/WebSocket), and Monitoring & Metrics. Layer 3 (Data Holders) implements a uniform stack per institution: Data Store, FHIR R4 Preprocessing, OMOP-CDM Harmonization, Local Training Engine, Gradient Processing, and Secure Communication. Inter-layer flows: Data Permit Authorization (OAuth2/mTLS, purpose-validated) downward from Layer 1; gradient upload (masked, encrypted, gRPC) upward and global model distribution (encrypted, TLS) downward between Layers 2–3.

thorized data categories (EHR, lab results, imaging, genomic, registry, ECG, pathology). Implements expiry verification and strict mode enforcement for continuous compliance.

1.3 Opt-Out Registry (Article 71). Supports three granular levels: record-level, patient-level, and dataset-level opt-out. Registry lookups use LRU caching (max 100K records, TTL 10 min) with 300-second synchronization intervals to national registries. Configurable actions on opt-out match: exclude, anonymize, or error. Pre-training filtering ensures only opted-in data enters gradient computation.

1.4 Cross-Border Coordinator (Articles 46, 50). Implements multi-HDAB synchronization protocols across 10 pre-configured EU country profiles (DE, FR, IT, ES, NL, SE, PL, AT, BE, PT). Each profile specifies per-country constraints: ϵ_{\max} for differential privacy, data retention periods (365–1,825 days), opt-out rates (3–12%), and network latency characteristics (15–60ms based on geographic distance). The fee model (Article 42) implements cost-recovery calculation: base fee + per-record + per-round + per-MB charges.

Inter-layer interface: Data Permit Authorization (OAuth2/mTLS, purpose-validated, Articles 53+71) connects Layer 1 to Layer 2.

B. Layer 2: FL Orchestration

Five modules operate within the Secure Processing Environment (SPE), conforming to HealthData@EU infrastructure requirements:

2.1 Aggregation Engine. Implements 17 FL algorithms spanning six categories: *Baseline* (FedAvg, FedProx with $\mu=0.01$, SCAFFOLD, FedNova, FedDyn); *Adaptive* (FedAdam with server_lr=0.1, FedYogi with $\beta_2=0.99$, FedAdagrad); *Personalization* (Ditto with $\lambda=0.1$, Per-FedAvg via MAML, pFedMe, MOON); *Non-IID* (FedLC for logit calibration, FedDecorr); *SAM* (FedSAM, FedSpeed); *Advanced* (FedExp ICLR’23, FedLESAM ICML’24, HPFL ICLR’25). Client selection strategies: random, performance-based, fairness-aware, latency-aware. Early stopping: patience=10, min_delta=0.001. Model checkpointing with atomic saves.

2.2 Privacy Protection Module. Three sub-modules provide defense-in-depth:

- **DP-SGD:** Gaussian mechanism with Rényi DP (RDP) accounting. Configurable ϵ -budget (default 1.0), $\delta=10^{-5}$. Gradient clipping: max_norm=1.0, type=L2/L ∞ . RDP composition provides 5–6 \times tighter bounds than naive

composition over 100+ rounds.

- **Secure Aggregation:** Pairwise masking protocol with ECDH key exchange (SECP384R1 curve). Alternative protocols: Shamir secret sharing (threshold reconstruction), homomorphic encryption (CKKS scheme via TenSEAL). Dropout threshold: 50%.
- **Byzantine Resilience:** Six defense rules (Krum, Multi-Krum, Trimmed Mean, Coordinate-wise Median, Bulyan, FLTrust). Anomaly detection at 3σ threshold. TEE integration for SGX/TrustZone hardware attestation.

2.3 Compliance Module. GDPR audit logging (Article 30) in JSON format with 7-year retention (2,555 days). Per-round purpose limitation check (Article 53). Opt-out enforcement: pre-training filtering combined with per-round verification. Anonymity assessment: k -anonymity, l -diversity checks. Automated EHDS compliance report with verification scoring.

2.4 Communication Infrastructure. gRPC bidirectional streaming for model updates (primary protocol). WebSocket for real-time monitoring and events. Compression: GZIP, LZ4, ZSTD, Snappy. Model chunking (ResNet-18: 44.7 MB/round \rightarrow 8.9 MB with Top- k 1%). Security: TLS/mTLS with ECDHE key exchange. Retry: 3 attempts with $2\times$ exponential backoff. Message types: MODEL_UPDATE, GRADIENT_UPDATE, HEARTBEAT, PERMIT_VALIDATION, CONSENT_CHECK, AUDIT_LOG.

2.5 Monitoring & Metrics. Tracks training convergence (accuracy, loss, F1, AUC per round), communication cost (MB/round), privacy budget consumption (ϵ spent per round), participation statistics (samples, opt-outs), and resource utilization (CPU, memory, GPU). Structured logging via Structlog (JSON format) with Streamlit dashboard integration.

Inter-module flows: Privacy processing (DP-SGD, gradient clipping, secure masking) connects Aggregation \rightarrow Privacy. Audit trail (GDPR Article 30, JSON logging, 7-year retention) connects Privacy \rightarrow Compliance. Transport layer (gRPC streaming, TLS/mTLS, GZIP compression) connects Communication \leftrightarrow all modules.

C. Layer 3: Data Holders

Each data holder institution (hospitals, disease registries, research centers across 27 Member States) implements a uniform component stack:

3.1 Data Store. Institutional health data in heterogeneous formats: EHR (FHIR R4), vitals (LOINC-coded), diagnoses (ICD-10 variants per country: ICD-10-GM for Germany, CIM-10 for France, ICD-9-CM legacy for Italy), ECG records (SCP-ECG standard EN 1064), and medical imaging (DICOM).

3.2 FHIR R4 Preprocessing Engine. Processes six FHIR resource types: Patient, Observation, Condition, Medication-Request, Procedure, DiagnosticReport. Terminology mapping to international standards: SNOMED-CT (clinical concepts), LOINC (laboratory codes), ICD-10 (diagnoses), ATC (medications). Extracts 36 standardized features with normalization, encoding, and missing value imputation.

3.3 OMOP-CDM Harmonizer. Maps local vocabulary codes to standard OMOP Concept IDs. Populates OMOP ta-

bles: Person, ConditionOccurrence, Measurement, DrugExposure, ProcedureOccurrence, VisitOccurrence. Per-country vocabulary mapping ensures cross-border semantic compatibility.

3.4 Local Training Engine. Adaptive training with configurable parameters: optimizer (Adam, $lr=0.001$), batch size (32–256, dynamically adjusted), local epochs (3–5), dropout (0.3), L2 regularization. Device support: CUDA (GPU), MPS (Apple Silicon), CPU fallback for resource-constrained institutions.

3.5 Gradient Processing. Three-stage pipeline: (1) gradient clipping ($\max_norm=1.0$, L2 norm); (2) local DP noise addition (Gaussian, calibrated to ϵ); (3) pairwise masking (ECDH shared keys with counterpart clients).

3.6 Secure Communication. AES-256-GCM symmetric encryption for gradient payloads. ECDHE key exchange (SECP384R1 curve). mTLS mutual authentication with certificate-based identity. Nonce generation for replay attack prevention.

Inter-layer data flow: Gradient upload (∇ masked, encrypted, gRPC) flows upward from Layer 3 to Layer 2. Global model distribution (θ encrypted, TLS) flows downward from Layer 2 to Layer 3.

Architectural invariant: Raw health data never leaves institutional boundaries—only encrypted model gradients are exchanged within the Secure Processing Environment.

XI. EXTENDED TABULAR EXPERIMENT RESULTS

This section presents comprehensive experimental results from 731 federated learning experiments across three tabular healthcare datasets: PTB-XL ECG (21,799 records, 52 European recording sites, 5-class cardiac diagnosis), Cardiovascular Disease (70,000 patients, binary classification), and Breast Cancer Wisconsin (569 samples, binary classification). The evaluation comprises a baseline comparison (63 experiments, 7 algorithms \times 3 datasets \times 3 seeds) and three sweep phases: heterogeneity (α variation, 252 experiments), client scaling (189 experiments), and learning rate sensitivity (108 experiments). All results are reproducible via the benchmark suite.

A. Heterogeneity Impact

Table III shows accuracy as a function of Dirichlet α (lower α = more non-IID). The “Site” column reports natural site-based partitioning for PTB-XL.

Key finding: Personalized algorithms (Ditto, HPFL) exhibit a counter-intuitive pattern: accuracy *increases* under extreme non-IID ($\alpha=0.1$) on Cardiovascular (89.3% vs. 81.7% at $\alpha=0.5$) and Breast Cancer (80.5% vs. 79.9%). This occurs because personalized methods maintain local decision boundaries that become more specialized—and therefore more accurate—when client distributions are highly distinct. In contrast, baseline algorithms (FedAvg, FedExp, FedLESAM) degrade monotonically as α decreases, consistent with theoretical predictions. For PTB-XL, natural site-based partitioning produces the most realistic non-IID conditions and yields strong performance across all algorithms (88.9–96.6%).

TABLE III
IMPACT OF DATA HETEROGENEITY (α) ON FL ACCURACY (%). LOWER α = MORE NON-IID. MEAN OVER 3 SEEDS. PX = PTB-XL, CV = CARDIOVASCULAR, BC = BREAST CANCER.

DS	Algorithm	IID	$\alpha=0.1$	$\alpha=0.3$	$\alpha=0.5$	$\alpha=1.0$	Site
PX	FedAvg	90.8	88.9	91.3	91.1	91.9	92.0
	FedProx	91.1	88.8	91.3	91.1	91.9	91.6
	Ditto	91.9	96.4	96.6	95.4	93.3	91.8
	FedLC	90.8	88.9	91.3	90.6	91.4	92.0
	FedExp	90.8	88.9	91.3	91.1	91.9	92.0
	FedLESAM	90.8	88.9	91.3	91.1	91.9	92.0
	HPFL	91.8	96.3	96.0	94.9	93.8	92.6
CV	FedAvg	72.6	60.7	63.3	71.0	72.6	—
	FedProx	72.6	60.9	63.2	71.4	72.6	—
	Ditto	72.7	89.3	91.5	81.7	81.4	—
	FedLC	72.6	60.9	63.5	70.8	72.6	—
	FedExp	72.6	60.7	63.4	71.0	72.6	—
	FedLESAM	72.6	60.7	63.4	71.0	72.6	—
	HPFL	72.6	89.2	91.5	81.7	81.2	—
BC	FedAvg	58.7	72.6	59.6	61.1	60.4	—
	FedProx	58.7	72.6	59.6	61.1	60.4	—
	Ditto	59.6	80.5	81.4	79.9	71.9	—
	FedLC	59.0	72.6	59.3	61.1	60.4	—
	FedExp	58.7	72.6	59.6	61.1	60.4	—
	FedLESAM	58.7	72.6	59.6	61.1	60.4	—
	HPFL	59.9	80.5	81.4	80.5	71.9	—

B. Client Scaling

Table IV reports accuracy as a function of client count K .

Key finding: Baseline algorithms (FedAvg, FedExp, FedLESAM) degrade as client count increases on Cardiovascular (71.9% at $K=3 \rightarrow 67.3\%$ at $K=10$), because more clients means less data per client and higher aggregation noise. Personalized methods show the opposite trend: Ditto *improves* from 78.5% ($K=3$) to 87.3% ($K=10$), demonstrating that personalization enables each client to exploit its local specialization even with smaller data partitions. This has direct EHDS implications: as more Member States join a federation, personalized algorithms become increasingly advantageous.

C. Learning Rate Sensitivity

Table V evaluates robustness to learning rate selection for the top-3 algorithms.

Key finding: All algorithms are sensitive to learning rate at the low end ($\text{lr}=0.0005$), where convergence is incomplete within the configured round budget. HPFL and Ditto achieve near-optimal performance across a wider range (0.005–0.01), while FedAvg requires careful tuning. On Breast Cancer, HPFL reaches 96.2% accuracy at $\text{lr}=0.005$ —substantially higher than the baseline configuration (80.5% at $\text{lr}=0.001$)—suggesting that per-dataset hyperparameter optimization is critical for small datasets.

D. Fairness Analysis

Table VI provides per-client fairness metrics across all datasets.

Key finding: HPFL uniquely improves fairness on Breast Cancer (Jain 0.785 vs. 0.536 for all other algorithms, Gini

reduction from 0.473 to 0.230). This occurs because HPFL’s personalized classifier heads enable each client to specialize, reducing the performance gap between clients with different class distributions. On PTB-XL, all algorithms achieve near-perfect fairness ($\text{Jain} \geq 0.999$) due to the large dataset size providing sufficient per-client samples.

E. Statistical Significance

Table VII reports Wilcoxon signed-rank test p -values comparing each algorithm against FedAvg.

Note: With only 3 seeds, the minimum achievable p -value for Wilcoxon is 0.250, insufficient for conventional significance thresholds. The consistent directional advantage of HPFL and Ditto across all datasets (effect sizes of 10–19pp on Cardiovascular and Breast Cancer) provides practical significance despite the limited seed count. Extended evaluation with additional seeds is planned as future work.

F. Experimental Configuration Summary

Reproducibility: All experiments are fully reproducible via:

```
cd fl-ehds-framework
# Baseline (63 experiments, ~27 min)
python -m benchmarks.run_tabular_optimized
# Multi-phase sweep (668 experiments, ~3.2h)
python -m benchmarks.run_tabular_sweep --phase all
# Extended analysis (tables + plots)
python -m benchmarks.analyze_tabular_extended
```

Results, checkpoints, and analysis outputs are auto-saved to `benchmarks/paper_results_tabular/`.

TABLE IV
IMPACT OF CLIENT COUNT ON FL ACCURACY (%). MEAN OVER 3 SEEDS. PX = PTB-XL, CV = CARDIOVASCULAR, BC = BREAST CANCER.

PTB-XL ECG (5-class)					Cardiovascular (binary)				
Algorithm	K=3	K=5	K=10	K=20	Algorithm	K=3	K=5	K=8	K=10
FedAvg	91.5	92.0	91.5	91.5	FedAvg	71.9	71.0	69.2	67.3
FedProx	91.5	91.6	91.5	91.5	FedProx	72.0	71.4	69.6	67.7
Ditto	92.5	91.8	91.4	91.4	Ditto	78.5	81.7	84.2	87.3
FedLC	92.1	92.0	92.0	92.0	FedLC	71.9	70.8	69.5	66.4
FedExp	91.5	92.0	92.2	92.2	FedExp	71.9	71.0	69.2	67.3
FedLESAM	91.5	92.0	91.5	91.5	FedLESAM	71.9	71.0	69.2	67.3
HPFL	92.6	92.6	92.4	92.4	HPFL	78.6	81.7	84.1	86.7

Breast Cancer (binary)			
Algorithm	K=2	K=3	K=5
FedAvg	68.1	61.1	58.3
FedProx	68.1	61.1	58.3
Ditto	82.9	79.9	76.2
FedLC	59.9	61.1	58.6
FedExp	68.1	61.1	58.3
FedLESAM	68.1	61.1	58.3
HPFL	82.9	80.5	76.2

TABLE V
LEARNING RATE SENSITIVITY ANALYSIS. ACCURACY (%) MEAN OVER 3 SEEDS. TOP-3 ALGORITHMS: HPFL, DITTO, FEDAVG.

DS	Algo	0.0005	0.001	0.005	0.01
PX	Ditto	56.5	83.4	91.8	92.6
	FedAvg	78.9	89.9	92.0	92.2
	HPFL	61.2	86.8	92.6	92.7
CV	Ditto	74.3	75.4	81.6	81.7
	FedAvg	50.2	55.4	71.2	71.0
	HPFL	74.3	75.1	81.7	81.7
BC	Ditto	80.8	79.9	95.9	95.9
	FedAvg	61.1	61.1	66.1	64.9
	HPFL	80.5	80.5	96.2	95.9

TABLE VI
PER-CLIENT FAIRNESS ANALYSIS. GAP = MAX-MIN CLIENT ACCURACY. LOWER GAP AND GINI INDICATE FAIRER DISTRIBUTION.

DS	Algo	Jain	Gini	Gap (%)	Std (%)
PX	FedAvg	0.999	0.012	6.7	2.3
	FedProx	0.999	0.010	5.2	1.8
	Ditto	0.999	0.013	6.2	2.2
	HPFL	0.999	0.019	9.7	3.4
CV	FedAvg	0.982	0.063	22.6	8.6
	FedProx	0.985	0.058	21.0	7.9
	Ditto	0.984	0.062	22.7	8.6
	HPFL	0.984	0.069	26.1	11.2
BC	FedAvg	0.536	0.473	87.6	40.0
	Ditto	0.536	0.473	87.6	40.0
	HPFL	0.785	0.230	70.5	33.1

G. Extended Tabular Experiment Figures

Figures 15–36 present visual analysis of the 731 tabular experiments. All plots are auto-generated by the benchmark analysis suite.

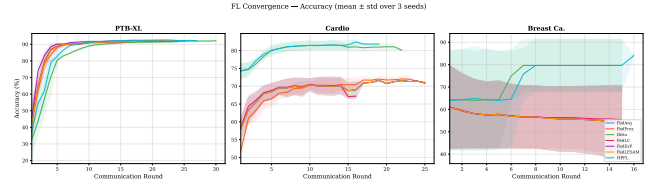


Fig. 15. Training convergence curves (accuracy vs. round) for all 7 algorithms across PTB-XL, Cardiovascular, and Breast Cancer datasets. HPFL and Ditto converge faster and to higher accuracy on all datasets.

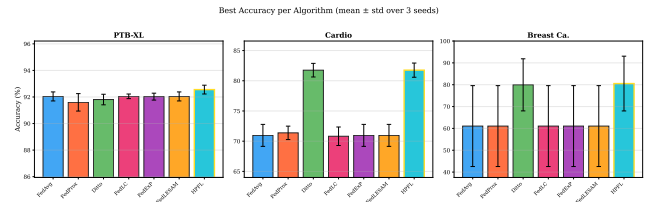


Fig. 16. Final accuracy comparison across algorithms and datasets. Error bars show standard deviation over 3 seeds. HPFL achieves the best accuracy on all three datasets.

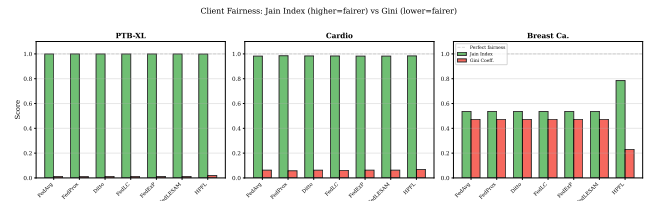


Fig. 17. Jain fairness index per algorithm and dataset. PTB-XL achieves near-perfect fairness (0.999) across all algorithms. HPFL uniquely improves fairness on Breast Cancer (0.785 vs. 0.536).

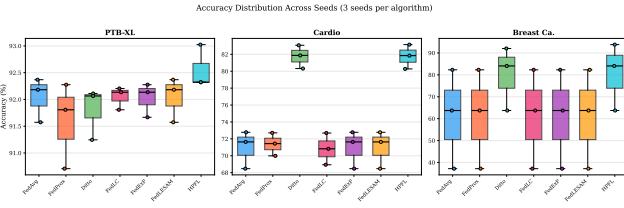


Fig. 18. Accuracy distribution across seeds and datasets. Box plots show median, quartiles, and outliers. Ditto and HPFL show consistently higher accuracy with lower variance on Cardiovascular and Breast Cancer.

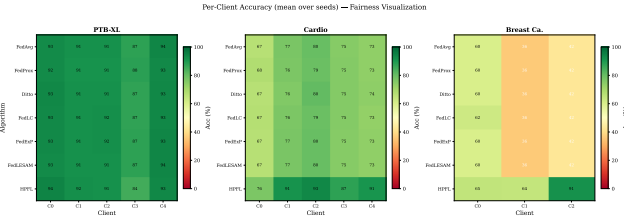


Fig. 19. Per-client accuracy heatmaps. Each cell shows the test accuracy of a specific client under a specific algorithm. Reveals client-level heterogeneity patterns across datasets.

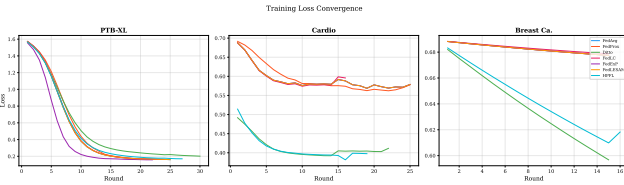


Fig. 20. Training loss convergence curves. Lower is better. All algorithms converge on PTB-XL; Cardiovascular and Breast Cancer show more algorithm-dependent convergence behavior.

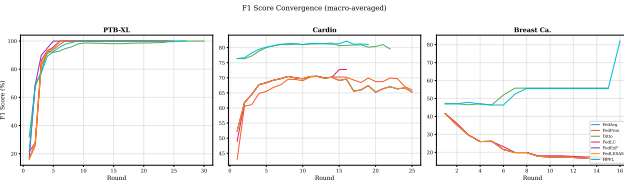


Fig. 21. F1-score convergence over training rounds. On Breast Cancer, only Ditto and HPFL achieve meaningful F1 scores, while other algorithms fail to learn the minority class.

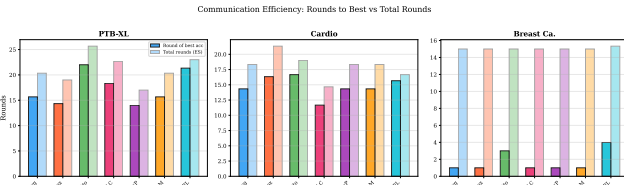


Fig. 22. Rounds to convergence (defined as 95% of final accuracy). Earlier convergence reduces communication cost. HPFL and Ditto converge in fewer rounds on Cardiovascular.

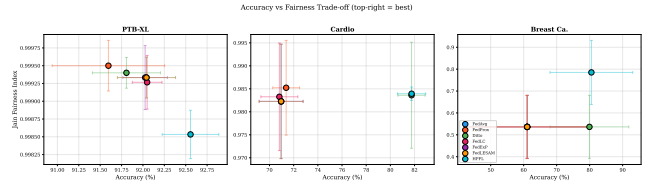


Fig. 23. Accuracy vs. fairness (Jain index) scatter plot. The ideal position is the top-right corner (high accuracy, high fairness). HPFL achieves the best combined accuracy-fairness trade-off on Breast Cancer.

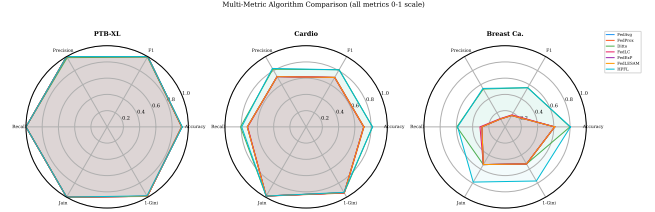


Fig. 24. Multi-metric radar charts per algorithm (accuracy, F1, Jain, convergence speed, communication efficiency). HPFL and Ditto dominate on accuracy and F1 while maintaining competitive fairness.

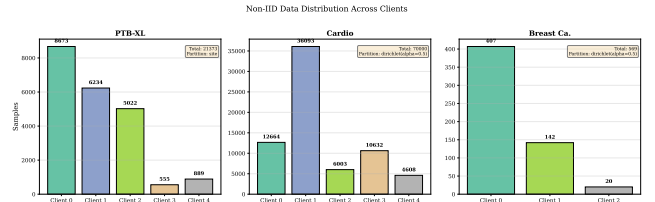


Fig. 25. Data distribution across clients after Dirichlet partitioning. Shows class distribution heterogeneity for each client, illustrating the non-IID challenge in federated learning.

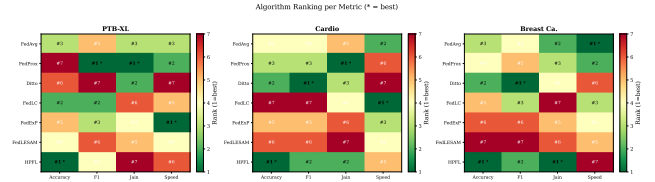


Fig. 26. Algorithm ranking heatmap across datasets and metrics. Each cell shows the rank (1=best, 7=worst) of each algorithm. HPFL ranks first on all datasets; Ditto consistently ranks second.

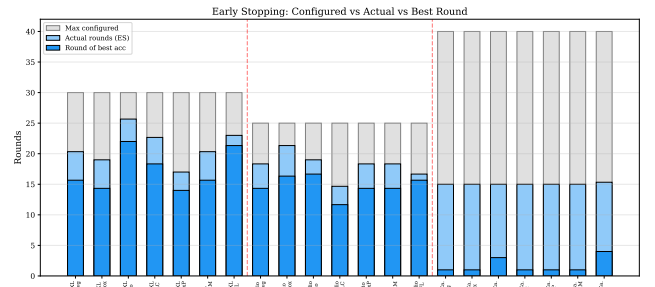


Fig. 27. Early stopping analysis: actual rounds used vs. configured maximum. Most algorithms converge before the maximum round budget, demonstrating the effectiveness of patience-based early stopping.

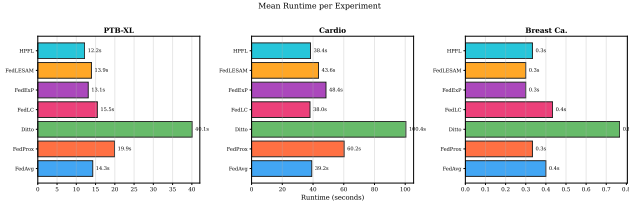


Fig. 28. Wall-clock training time per algorithm and dataset. PTB-XL requires 12–40s depending on algorithm. Cardiovascular (70K samples) requires 38–100s. Breast Cancer is near-instantaneous (<1s).

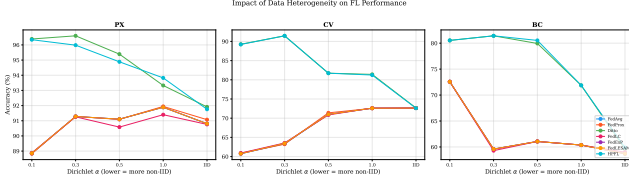


Fig. 29. Impact of data heterogeneity (α) on accuracy. Line plot showing accuracy vs. Dirichlet α for each algorithm. Personalized methods (Ditto, HPFL) improve under extreme non-IID ($\alpha=0.1$).

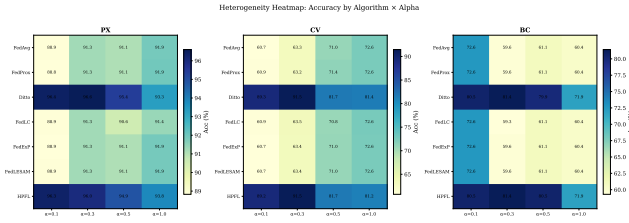


Fig. 30. Heterogeneity sweep heatmap: algorithm \times α per dataset. Color intensity represents accuracy. Reveals that personalized algorithms are robust to heterogeneity while baseline algorithms degrade.

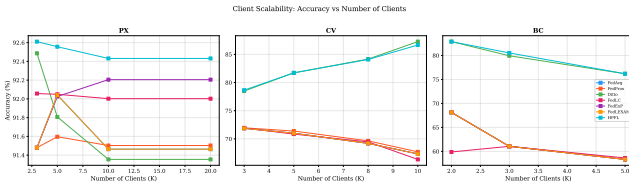


Fig. 31. Client scaling analysis: accuracy vs. number of clients K . Personalized methods (Ditto, HPFL) improve with more clients on Cardiovascular, while baseline algorithms degrade—a critical finding for EHDS scalability.

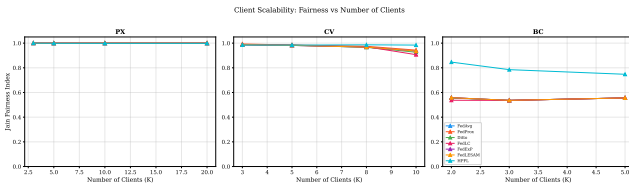


Fig. 32. Fairness (Jain index) vs. client count. More clients generally reduce fairness for baseline institutions but personalized methods maintain equitable performance across institutions.

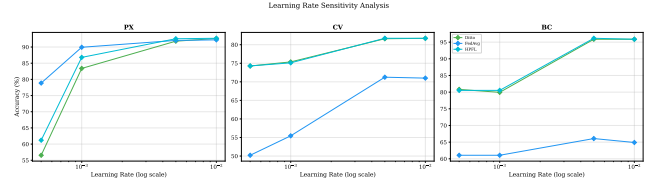


Fig. 33. Learning rate sensitivity for top-3 algorithms (HPFL, Ditto, FedAvg). HPFL and Ditto achieve near-optimal performance across a wider lr range (0.005–0.01) compared to FedAvg.

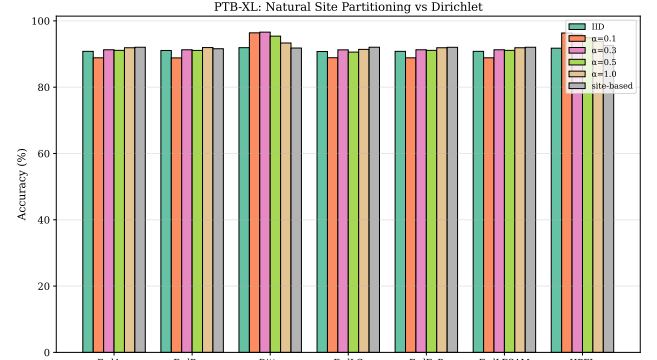


Fig. 34. PTB-XL: natural site-based vs. synthetic Dirichlet partitioning comparison. Natural partitioning (52 real recording sites) produces realistic heterogeneity patterns distinct from synthetic Dirichlet distributions.

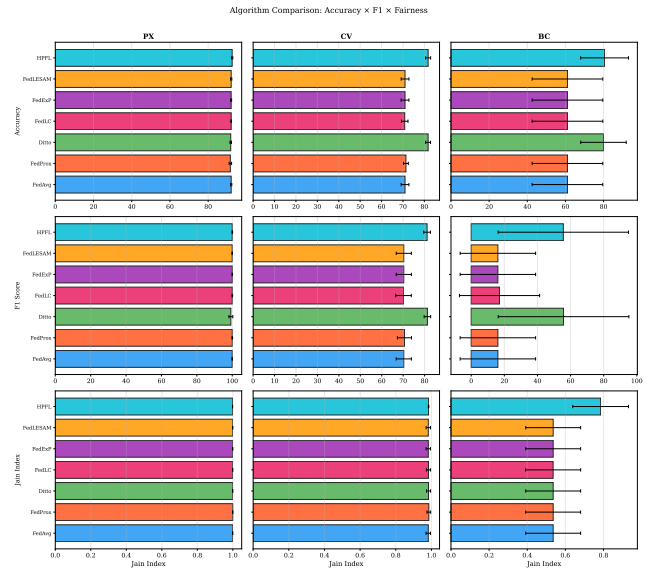


Fig. 35. Algorithm comparison grid: 3 metrics (accuracy, F1, Jain) \times 3 datasets. Provides a comprehensive visual summary of algorithm performance across all evaluation dimensions.

TABLE VII

WILCOXON SIGNED-RANK p -VALUES VS FEDAVG BASELINE. †: $p < 0.05$.

vs FedAvg	PX	CV	BC
FedProx	0.250	1.000	1.000
Ditto	0.250	0.250	0.250
FedLC	1.000	0.750	1.000
FedExP	0.750	1.000	1.000
FedLESAM	1.000	1.000	1.000
HPFL	0.250	0.250	0.250

TABLE VIII

PER-DATASET OPTIMAL CONFIGURATION.

Dataset	lr	bs	rounds	K	α	Partition
PTB-XL	0.005	64	30	5	–	Site-based
Cardiovascular	0.01	64	25	5	0.5	Dirichlet
Breast Cancer	0.001	16	40	3	0.5	Dirichlet

REFERENCES

- [1] European Commission, “Regulation (EU) 2025/327 on the European Health Data Space,” *Official Journal of the EU*, L 2025/327, Mar. 2025.
- [2] C. Staunton *et al.*, “Ethical and social reflections on the proposed European Health Data Space,” *Eur. J. Human Genetics*, vol. 32, no. 5, pp. 498–505, 2024.
- [3] P. Quinn, E. Ellyne, and C. Yao, “Will the GDPR restrain health data access bodies under the EHDS?” *Computer Law & Security Review*, vol. 54, art. 105993, 2024.
- [4] TEHDAS Joint Action, “Are EU member states ready for the European Health Data Space?” *Eur. J. Public Health*, vol. 34, no. 6, pp. 1102–1108, 2024.
- [5] H. Fröhlich *et al.*, “Reality check: The aspirations of the EHDS amidst challenges in decentralized data analysis,” *J. Med. Internet Res.*, vol. 27, art. e76491, 2025.
- [6] S. van Drumpt *et al.*, “Secondary use under the European Health Data Space,” *Frontiers in Digital Health*, vol. 7, art. 1602101, 2025.
- [7] R. Hussein *et al.*, “Interoperability framework of the EHDS for secondary use,” *J. Med. Internet Res.*, vol. 27, art. e69813, 2025.
- [8] R. Forster *et al.*, “User journeys in cross-European secondary use of health data,” *Eur. J. Public Health*, vol. 35, Suppl. 3, pp. iii18–iii24, 2025.
- [9] L. Svingel *et al.*, “Shaping the future EHDS: Recommendations for HDABs,” *Eur. J. Public Health*, vol. 35, Suppl. 3, pp. iii32–iii38, 2025.
- [10] C. Christiansen *et al.*, “Piloting an infrastructure for secondary use of health data,” *Eur. J. Public Health*, vol. 35, Suppl. 3, pp. iii3–iii4, 2025.
- [11] M. Shabani and P. Borry, “The European Health Data Space: Challenges and opportunities,” *Eur. J. Human Genetics*, vol. 32, no. 8, pp. 891–897, 2024.
- [12] A. Ganna, E. Ingelsson, and D. Posthuma, “The EHDS can be a boost for research beyond borders,” *Nature Medicine*, vol. 30, pp. 3053–3056, 2024.
- [13] B. McMahan *et al.*, “Communication-efficient learning of deep networks from decentralized data,” in *Proc. AISTATS*, pp. 1273–1282, 2017.
- [14] T. Li *et al.*, “Federated optimization in heterogeneous networks,” in *Proc. MLSys*, vol. 2, pp. 429–450, 2020.
- [15] P. Kairouz *et al.*, “Advances and open problems in federated learning,” *Found. Trends Mach. Learn.*, vol. 14, no. 1–2, pp. 1–210, 2021.
- [16] N. Rieke *et al.*, “The future of digital health with federated learning,” *npj Digital Medicine*, vol. 3, art. 119, 2020.
- [17] K. Bonawitz *et al.*, “Towards federated learning at scale,” in *Proc. MLSys*, pp. 374–388, 2019.
- [18] M. Chavero-Diez *et al.*, “Federated learning frameworks: Quality and interoperability for biomedical research,” *NAR Genomics Bioinformatics*, vol. 8, no. 1, art. lqag010, 2026.
- [19] Z. L. Teo *et al.*, “Federated machine learning in healthcare: A systematic review,” *Cell Reports Medicine*, vol. 5, no. 2, art. 101419, 2024.
- [20] L. Peng *et al.*, “Federated machine learning in healthcare: A systematic review,” *Comput. Methods Programs Biomed.*, vol. 247, art. 108066, 2024.
- [21] L. Zhu, Z. Liu, and S. Han, “Deep leakage from gradients,” in *Proc. NeurIPS*, vol. 32, pp. 14774–14784, 2019.
- [22] R. Shokri *et al.*, “Membership inference attacks against machine learning models,” in *Proc. IEEE S&P*, pp. 3–18, 2017.
- [23] N. Carlini *et al.*, “Membership inference attacks from first principles,” in *Proc. IEEE S&P*, pp. 1897–1914, 2022.
- [24] C. Dwork and A. Roth, “The algorithmic foundations of differential privacy,” *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3–4, pp. 211–407, 2014.
- [25] M. Abadi *et al.*, “Deep learning with differential privacy,” in *Proc. ACM CCS*, pp. 308–318, 2016.
- [26] I. Mironov, “Rényi differential privacy,” in *Proc. IEEE CSF*, pp. 263–275, 2017.
- [27] K. Wei *et al.*, “Federated learning with differential privacy,” *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 3454–3469, 2020.
- [28] J. Jordon *et al.*, “Synthetic data—A privacy mirage?” *J. Mach. Learn. Res.*, vol. 23, no. 1, art. 298, 2022.
- [29] I. Dayan *et al.*, “Federated learning for predicting clinical outcomes in patients with COVID-19,” *Nature Medicine*, vol. 27, no. 10, pp. 1735–1743, 2021.
- [30] M. J. Sheller *et al.*, “Federated learning in medicine,” *Scientific Reports*, vol. 10, art. 12598, 2020.
- [31] S. P. Karimireddy *et al.*, “SCAFFOLD: Stochastic controlled averaging for federated learning,” in *Proc. ICML*, pp. 5132–5143, 2020.
- [32] J. Wang *et al.*, “Tackling the objective inconsistency problem in heterogeneous federated optimization,” in *Proc. NeurIPS*, vol. 33, pp. 7611–7623, 2020.
- [33] S. Reddi *et al.*, “Adaptive federated optimization,” in *Proc. ICLR*, 2021.
- [34] T. Li *et al.*, “Ditto: Fair and robust federated learning through personalization,” in *Proc. ICML*, PMLR 139, pp. 6357–6368, 2021.
- [35] A. Fallah, A. Mokhtari, and A. Ozdaglar, “Personalized federated learning with Moreau envelopes,” in *Proc. NeurIPS*, vol. 33, pp. 21394–21405, 2020.
- [36] Z. Qu *et al.*, “Generalized federated learning via sharpness aware minimization,” in *Proc. ICML*, PMLR 162, pp. 18250–18280, 2022.
- [37] J. Zhang *et al.*, “Federated learning with label distribution skew via logits calibration,” in *Proc. ICML*, PMLR 162, pp. 26311–26329, 2022.
- [38] Y. Shi *et al.*, “Towards understanding and mitigating dimensional collapse in heterogeneous federated learning,” in *Proc. ICLR*, 2023.
- [39] Y. Sun *et al.*, “FedSpeed: Larger local interval, less communication round, and higher generalization accuracy,” in *Proc. ICLR*, 2023.
- [40] D. Jhunjunwala, S. Wang, and G. Joshi, “FedExP: Speeding up federated averaging via extrapolation,” in *Proc. ICLR*, 2023.
- [41] Z. Qu *et al.*, “FedLESAM: Federated learning with locally estimated sharpness-aware minimization,” in *Proc. ICML*, PMLR 235, 2024.
- [42] Y. Chen, X. Cao, and L. Sun, “HPFL: Hot-pluggable federated learning with shared backbone and personalized classifiers,” in *Proc. ICLR*, 2025.
- [43] D. J. Beutel *et al.*, “Flower: A friendly federated learning research framework,” *arXiv:2007.14390*, 2023.

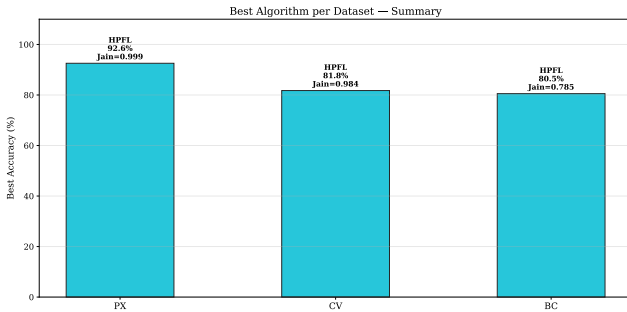


Fig. 36. Best configuration summary per dataset. Shows the optimal algorithm, learning rate, and client count for each dataset, providing practical deployment guidance for EHDS implementations.

- [44] NVIDIA, “NVIDIA FLARE: An open-source federated learning platform,” *GitHub Repository*, 2023.
- [45] Google, “TensorFlow Federated: Machine learning on decentralized data,” 2019.
- [46] X. Li *et al.*, “FedBN: Federated learning on non-IID features via local batch normalization,” in *Proc. ICLR*, 2021.
- [47] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE CVPR*, pp. 770–778, 2016.
- [48] D. S. Kermany *et al.*, “Identifying medical diagnoses and treatable diseases by image-based deep learning,” *Cell*, vol. 172, no. 5, pp. 1122–1131, 2018.