IMAGE ORCHESTRATION · CAPABILITY MATRIX

One capability set, purpose-built products.

The day-to-day imaging fleet, delivered across focused applications rather than one overloaded router — every mark below **verified against our own source code**, not a datasheet. The goal is **more, but not less**: meet the competitive baseline and add the depth a single-product tier ladder can't show — marked ★.

`7` products · `35` capabilities · Code-verified `2026-09-09` · Non-device · §520(o)(1)(D)

● **Full** — implemented & code-verified ◐ **Partial** — one-directional or limited scope — **Not present** — by design or absent
★ **Fleet baseline** — depth a tier ladder doesn't list

HOW THE TIERS MAP TO A SINGLE-PRODUCT LADDER

Where the incumbent approach sells one product in five ascending tiers, we ship purpose-built applications — so a capability the routers don't carry appears in its dedicated product's column instead of a greyed-out tier. The four ★ rows — encryption at rest, enterprise auth, a hash-chained HIPAA audit trail, and configuration backup & DR — are baseline across our whole fleet, on every product regardless of tier.

Streaming tier → **Essential Router**	Store-and-forward tier → **Core Router**
Full / professional tier → **XyDromatics Router**	Enterprise / edge tier → **Core Router + XyDromatics Router**
Image-exchange tier → **SynthCloudConnect**	— no equivalent tier — → **Encounter Engine · Core MWL · SR Engine**

CAPABILITY	Essential Router STREAMING	Core Router STORE & FORWARD	Core MWL WORKLIST	XyDromatics Router FULL PLATFORM	SR Engine REPORTING	Encounter Engine ORDER WORKFLOW	SynthCloudConnect CROSS-SITE RELAY
CORE ROUTING & TRANSFORMATION							
DICOM DIMSE image routing C-STORE SCP / SCU	●	●	—	●	◐ [1]	●	—
Configurable, priority-based routing rules	●	●	—	●	●	—	—
Tag-morphing & filtering	— [2]	●	—	●	—	●	—
Order reconciliation study tags ⇔ authoritative order (local orders or an external Modality Worklist — no HL7 needed) · identity/laterality flagged	—	● [15]	—	●	—	—	—
De-identification & anonymization metadata + burned-in pixel	—	●	—	●	◐ [3]	◐ [3]	—
Compression / transfer-syntax conversion	—	●	—	●	—	—	—
Multiplex routing to archives / VNAs cloud + on-prem	●	●	—	●	●	●	—

CAPABILITY	Essential Router STREAMING	Core Router STORE & FORWARD	Core MWL WORKLIST	XyDromatics Router FULL PLATFORM	SR Engine REPORTING	Encounter Engine ORDER WORKFLOW	SynthCloudConnect CROSS-SITE RELAY
OPERATIONAL DEPTH							
Store-and-forward durability durable spool · retry · dead-letter	—	●	—	●	●	●	◐ 4
Load-aware routing & destination failover	—	●	—	●	●	◐ 5	—
Load-balanced destination pools study-affinity sending · per-member health & failover	—	●	—	●	—	—	—
Hold queue / manual-review gate	—	●	—	●	—	●	—
Queryable study cache local C-FIND / C-MOVE + QIDO / WADO	—	● 6	—	●	—	—	—
Passive device / modality inventory auto-harvested equipment identity · reachability	—	—	—	● 14	—	—	—
ORDERS & WORKLIST							
HL7 messaging (ingest / route) ORM · ORU · ADT	—	—	●	●	● 7	●	—
Provide Modality Worklist (MWL C-FIND SCP) serve worklist from HL7 orders	—	—	●	●	—	—	—
MWL query proxy / relay to upstream RIS proxy · merge with local worklist	—	—	●	●	—	—	—
Order generation from inbound exams e.g. POCUS	—	—	—	●	—	●	—
SECURITY, EXCHANGE & COMPLIANCE							
In-transit security (DICOM-TLS / mTLS)	●	●	●	●	●	◐ 8	●
★ Encryption at rest customer-keyed key-escrow	●	●	●	●	●	●	●
★ Enterprise auth (LDAP + RBAC / custom roles)	●	●	●	●	●	●	◐ 9
★ HIPAA audit trail hash-chained · 6-yr retention	●	●	●	●	●	●	●
DICOM ↔ DICOMweb (STOW / QIDO / WADO)	—	● 13	—	●	◐ 10	—	● 11
Cross-site / cloud image exchange outbound-initiated	—	—	—	●	—	—	●
AI workflow & cloud result delivery	—	—	—	●	—	—	—
REPORTING, ANALYTICS & SUPPORT							
Report rendering / file conversion SR · encapsulated PDF → PDF	—	—	—	◐ 12	—	—	—
DICOM SR & dictation / report integration	—	—	—	●	●	—	—
Dose-monitoring integration	—	—	—	●	●	—	—
Analytics integration (SynthInSight)	—	●	●	●	●	●	—
Query/Retrieve prefetch (pull priors) C-FIND / C-MOVE	—	—	—	●	—	—	—
★ Configuration backup & DR sealed config-sync hub	●	●	●	●	●	●	●
Remote support & telemetry via SynthGateway	●	●	●	●	●	●	●
Fleet console management via SynthConstellation	●	●	●	●	●	●	●

CAPABILITY	Essential Router STREAMING	Core Router STORE & FORWARD	Core MWL WORKLIST	XyDromatics Router FULL PLATFORM	SR Engine REPORTING	Encounter Engine ORDER WORKFLOW	SynthCloudConnect CROSS-SITE RELAY
DEPLOYMENT & PLATFORM							
Windows Server (MSI installer)	●	●	●	●	●	●	●
Linux (self-installing tarball + systemd)	●	●	●	●	●	●	●
Virtual appliance (OVA / VHDX / VMDK) turnkey VM image	●	●	●	●	●	●	●

● code-verified full · ◐ partial · — not present · ★ fleet baseline · marks from source, not marketing.

ABOVE THE GRID

The fleet-operations layer.

Two products sit *above* the router matrix — the load-balancing and fleet-console layers the incumbents sell as separate SKUs, answered head-on. Neither is a peer to a router: SynthIQ fronts any DICOM backend; SynthConstellation runs the whole XyDromatics fleet from one board.

SynthIQ

High availability & business continuity — for any DICOM fleet, not just ours

PATENT PENDING · 2 US APPLICATIONS

Competitors fold load balancing into their router tiers. We ship it as a distinct, **vendor-neutral** product — **SynthIQ** is our **high-availability & business-continuity** layer in front of **any** DICOM-conformant backend: a **heterogeneous, multi-vendor** pool of our routers **and** third-party PACS, VNAs, and cloud endpoints — not one vendor's boxes. It balances **two lanes** — DICOM image traffic (C-STORE, on persistent **Study-Instance-UID affinity** so every study stays on one backend across associations, follow-ups days later, and restarts) **and** Modality Worklist queries (MWL C-FIND, health-aware) — and its three-tier waterfall (affinity → least-busy → fallback) keeps imaging flowing when a backend goes down. That's why it isn't a column in the matrix: it's the layer *above* the grid.

PROVIDES
High availability + business continuity

BALANCES
DICOM (C-STORE) + Modality Worklist (MWL C-FIND)

BACKENDS
Any DICOM router — heterogeneous, multi-vendor

HEALTH
JSON API for integrators; SynthPulse black-box probing for backends that can't / won't

DEPLOYS
Windows · Linux · OVA/VHDX/VMDK

SynthConstellation

One console for the whole fleet

The incumbent sells fleet monitoring as a separate SKU. We ship **SynthConstellation** — a single pane of glass that pulls **every product's** health, queues, and config-sync posture into one board: routers, engines, archives — the **whole fleet, not just the routers**. Open any product's own console right from it with **single sign-on**, and run the fleet from one place instead of logging into each. It also hosts the customer-controlled **Config Backup Hub**: every enrolled product seals its configuration on a schedule, so a lost or replaced host reseeds its exact config for disaster recovery. **Operational metadata only** — it moves no images and holds no PHI.

SEE
Health · queue depth · config posture — every product in the fleet

REACH
Open any product's console via single sign-on

RECOVER
Config Backup Hub — sealed config → reseed for DR

SCOPE
Whole fleet, not just routers · metadata only (no images, no PHI)

EVIDENCE & SCOPE

1. SR Engine receives via DIMSE C-STORE SCP (+ C-ECHO) but forwards over STOW-RS — no C-STORE SCU.
2. Essential Router does object filtering (drop rules, per-source allowed-modalities) but no tag morphing / coercion — streaming by design.
3. Metadata pseudonymization only (reversible, keyed); no burned-in-pixel redaction on these products.
4. SynthCloudConnect holds studies in a transit spool with a TTL (bridge relay), not a durable archive.
5. Encounter Engine emits a load-aware health score for the SynthIQ pool, but has no internal destination failover.
6. Core Router: a bounded, transient, sealed local Query/Retrieve cache of recently-received studies — DIMSE C-FIND / C-MOVE plus QIDO-RS / WADO-RS — so a site with no archive can run Query/Retrieve without a VNA. A cache, not an archive: TTL / size eviction is a hard delete.
7. SR Engine routes HL7 ORU^R01 over MLLP; it does not serve a Modality Worklist.
8. Encounter Engine uses server-side TLS only; no mutual-TLS. (No product implements OAuth2 — see below.)
9. SynthCloudConnect authenticates via identity-provider / OTP access (Cloudflare Access), not LDAP; RBAC and custom roles still apply.
10. Outbound STOW-RS delivery only (a DICOMweb destination) — not a full bidirectional STOW / QIDO / WADO proxy.
11. SynthCloudConnect exposes full STOW-RS / QIDO-RS / WADO-RS over the bridge, but no DIMSE C-STORE.
12. The flagship renders / extracts DICOM SR & encapsulated-PDF report objects to PDF; it does not author encapsulated-PDF / SR objects.
13. Core Router serves QIDO-RS + WADO-RS (read) from its bounded transient study cache, delivers outbound STOW-RS to DICOMweb destinations, and accepts inbound STOW-RS ingest — a token / mutual-TLS machine door that feeds each pushed instance into the same routing pipeline as a C-STORE.
14. The flagship passively inventories every device that connects — PHI-free equipment identity (manufacturer, model, station, modality) plus on-demand C-ECHO reachability — license-gated. Essential Router, Core Router and Core MWL sites get the same capability from the standalone XyDromatics Device Registry product.
15. Core Router has no local HL7 order store, so it reconciles a received study's tags against an external provider's Modality Worklist — fetched by an initiating C-FIND — with no HL7 ingest. The flagship reconciles against its own HL7 order store and can additionally fall back to an external worklist for accessions the local store does not have.

READING THE MARKS

● **Full** means implemented and verified in the shipping source — not a label on a spec sheet. ◐ **Partial** flags a real but one-directional or limited implementation, footnoted so you know exactly where the edge is. **—** means the capability lives in a different product on purpose, so you buy the tool that does it well.

Transport security is TLS + mutual-TLS with cert pinning — there is no OAuth2 / OIDC anywhere in the fleet, so we lead with mTLS rather than claim an OAuth2 checkbox we don't implement.

Where a capability is only **partial** on a tier, we mark it ◐ and footnote exactly what is and isn't there rather than claim a full checkbox. Every ● is verified against our own source code — if a mark isn't solid, we say so.