

## **HACIA REPOSITARIOS CLÍNICOS OPENEHR A GRAN ESCALA: RESULTADOS Y LECCIONES APRENDIDAS EN CATALUÑA**

J. PIERA JIMÉNEZ<sup>1,2</sup>, V. MÜLLER<sup>3</sup>, H. REISE<sup>3</sup>, A. VIDREAN<sup>3</sup>, P. BEIRÁN AMIGO<sup>4</sup>, J. BESCÓS GRILLO<sup>4</sup>, J.A. MIRA PALACIOS<sup>5</sup>, J. GABALDÀ AZOFRA<sup>5</sup>

<sup>1</sup>*Dirección de Sistemas de Información, Servei Català de la Salut, Barcelona, España*

<sup>2</sup>*Grupo de investigación Digitalización para la Sostenibilidad del Sistema Sanitario (DS3), Barcelona, España*

<sup>3</sup>*vitagroup health intelligence GmbH, Mannheim, Alemania*

<sup>4</sup>*IBM Consulting, Barcelona, España*

<sup>5</sup>*Generalitat de Catalunya – CTTI, Barcelona, España*

### **INTRODUCCIÓN**

Al igual que en la mayoría de los países occidentales, la transformación digital en Cataluña se ha convertido en un elemento esencial para sostener y mejorar el sistema sanitario catalán [1,2]. Esta necesidad está impulsada por el envejecimiento demográfico, el aumento de las enfermedades crónicas, los retos de sostenibilidad y las crecientes expectativas de la ciudadanía [1]. Los sistemas de información sanitaria pueden constituir un motor clave de dicha transformación digital. No obstante, es ampliamente reconocido que también pueden convertirse en barreras cuando permanecen fragmentados, presentan bajo rendimiento y carecen de interoperabilidad adecuada [3,4].

Una crítica relevante a los sistemas actuales es que muchos de ellos se basan en un modelo asistencial episódico, centrado en la actividad y organizado en silos. Esto genera islas de información, duplicidad documental y una experiencia deficiente tanto para los profesionales sanitarios como para los ciudadanos [1–4]. Para abordar estos desafíos técnicos y sociales, el Departament de Salut de Cataluña ha diseñado la Plataforma Oberta (Open Platform), orientada a rediseñar los flujos asistenciales en torno a una atención longitudinal, integrada y centrada en la persona [5].

Un requisito fundamental para el desarrollo de la Plataforma Oberta es disponer de un modelo de información semánticamente rico y expresivo que represente adecuadamente el recorrido del paciente a lo largo del sistema sanitario. Con este objetivo se seleccionó el estándar abierto openEHR, que permite la representación unificada de la información clínica en un Repositorio Clínico de Datos (Clinical Data Repository, CDR), concebido como la columna vertebral de la plataforma.

El CDR basado en openEHR utilizado en la Plataforma Oberta es el Health Intelligence Platform (HIP) (Vitagroup Health Intelligence GmbH, Mannheim, Alemania), que opera sobre una base de datos relacional distribuida (Yugabyte v2025.2).

Aunque los repositorios openEHR existen desde hace casi 30 años, ninguna implementación previa había gestionado datos sanitarios a la escala poblacional de Cataluña. Además, el modelo centrado en el paciente que impone openEHR tiene implicaciones directas en el diseño de sistemas de Procesamiento de Transacciones en Línea (OLTP). Los CDR openEHR están diseñados para operaciones CRUD altamente concurrentes, focalizadas en un paciente cada vez, manteniendo cumplimiento estricto de las propiedades ACID y versionado. Esto requiere múltiples operaciones de join entre arquetipos referenciados en la cláusula contains, acceso a elementos en rutas profundas y navegación de relaciones complejas inherentes al Modelo de Referencia openEHR.

Estas características plantean desafíos significativos para alcanzar alto rendimiento en consultas mediante el lenguaje Archetype Query Language (AQL) a la escala requerida por la Plataforma Oberta. Durante la implementación del CDR, estas limitaciones de rendimiento en AQL hicieron necesarias amplias estrategias de optimización.

## **OBJETIVO**

Durante los últimos dos años se han implementado diversos mecanismos de optimización del CDR orientados a consultas AQL vinculadas al paciente. El objetivo de este trabajo es presentar el despliegue del CDR HIP en la Plataforma Oberta, los desafíos encontrados y las mejoras introducidas para soportar una implementación a escala regional (13 millones de pacientes).

En primer lugar, se describen los métodos utilizados para optimizar el procesador AQL del HIP. En segundo lugar, se presentan los métodos de evaluación. En tercer lugar, se exponen los resultados obtenidos mediante las estrategias de optimización. Finalmente, se discute qué enfoques resultan más adecuados según los distintos escenarios.

## MÉTODOS

Se evaluaron tres estrategias principales de optimización: (1) procesamiento AQL en memoria (In-Memory Projection, IMP), (2) estrategias de indexación convencionales e integradas mediante tablas índice, y (3) AQL manual mediante SQL personalizado. Las evaluaciones se realizaron sobre un subconjunto anonimizado del 10% del entorno productivo, considerando escenarios con una y diez solicitudes concurrentes.

### *3.1. AQL en memoria (In Memory Projection)*

Los motores AQL traducen consultas AQL al lenguaje de consulta subyacente de la base de datos (por ejemplo, SQL en bases relacionales) y delegan su ejecución en el motor de base de datos. Posteriormente, los resultados son reconstruidos en un formato conforme a openEHR. Este enfoque resulta ineficiente cuando se trata de rutas largas que requieren múltiples operaciones de join para preservar la semántica prevista.

Para superar esta limitación se desarrolló una técnica de AQL en memoria (In Memory Projection, IMP), que ejecuta parte de la lógica de la consulta en la memoria de la aplicación tras recuperar los datos de la base de datos. Este método consta de dos fases: (i) recuperación de un conjunto de resultados más amplio o simplificado desde la base de datos subyacente y (ii) ejecución en memoria de proyección, filtrado, agregación y ordenación sobre estructuras openEHR.

Las pruebas se realizaron sobre un subconjunto anonimizado correspondiente al 10% de las historias clínicas completas del entorno productivo, evaluando escenarios con una y diez solicitudes concurrentes.

### *3.2. Estrategia de indexación*

Una segunda estrategia para mejorar la ejecución de AQL consistió en el desarrollo de estructuras de indexación diseñadas específicamente para materializar rutas de openEHR consultadas con frecuencia en estructuras relacionales. La estrategia de indexación se centró en dos mecanismos complementarios:

#### *3.2.1. Índices convencionales*

Se definieron índices sobre atributos temporales de alto uso (ACTION.time, POINT\_EVENT.time, INTERVAL\_EVENT.time, EVENT\_CONTEXT.start\_time) y sobre campos de composición para optimizar rutas de consulta. Asimismo, se crearon índices adicionales sobre los datos de las composiciones con el fin de mejorar el rendimiento de las rutas de consulta y facilitar optimizaciones orientadas a la eliminación de operaciones de join.

### *3.2.2. Enfoque integrado (tabla índice)*

Otra estrategia de indexación propuesta fue la implementación de una tabla índice diseñada para dar soporte a casos de uso específicos. Esta estrategia tuvo en consideración los impactos operativos, incluido el riesgo de requerir una tabla índice dedicada para cada caso de uso. Cuando resultó pertinente, el equipo también analizó la mecánica de los índices y el comportamiento de los planes de ejecución de consultas con el fin de garantizar mejoras de rendimiento. La creación de índices se planificó y ejecutó bajo restricciones operativas explícitas. El impacto de la indexación se evaluó mediante ciclos repetidos de pruebas de rendimiento, que incluyeron:

- Ejecuciones automáticas de pruebas de rendimiento tras la introducción de nuevos índices y el ajuste de consultas AQL, con seguimiento explícito de la latencia en escenarios de mejor y peor caso bajo distintas restricciones de intervalo temporal.
- Simulación del entorno mediante la restauración de estadísticas de PREPROD/PROD en un entorno en la nube (utilizando un subconjunto anonimizado que contenía el 10 % del conjunto de datos productivo anonimizado con historias clínicas completas), con el objetivo de reproducir el comportamiento del planificador de consultas y permitir la comparación del rendimiento de cargas SQL y AQL.
- Monitorización operativa durante migraciones y despliegues escalonados, incluyendo la observación del uso de disco y de los cambios de tamaño asociados a la indexación.

### *3.3. AQL manual*

El AQL manual es una funcionalidad diseñada para proporcionar instrucciones SQL personalizadas en lugar de depender del motor AQL para traducir la consulta AQL. Aprovechando un conocimiento profundo del modelo de datos y de las propiedades de la base de datos, a menudo es posible diseñar consultas SQL que superen en rendimiento al motor AQL. Estas consultas SQL personalizadas pueden incorporarse como parte de la definición de una consulta AQL almacenada. Este enfoque permite un ajuste fino del rendimiento de AQL, manteniendo al mismo tiempo la interfaz y la parametrización a nivel AQL utilizadas por la aplicación.

## **RESULTADOS**

A lo largo del período de evaluación, el despliegue demostró la viabilidad funcional para la ingestión de datos openEHR a gran escala y la continuidad operativa al seguir determinados patrones en el diseño de consultas AQL, el uso de AQL en memoria y la aplicación de estrategias de indexación. Los hallazgos más relevantes se describen en las secciones siguientes.

#### *4.1.AQL en memoria*

La evaluación comparativa de IMP se centró exclusivamente en consultas de un único paciente (una historia clínica openEHR). En el conjunto de consultas AQL analizadas, la activación del AQL en memoria produjo una reducción sustancial de la latencia mediana en comparación con la vía de ejecución convencional. En condiciones sin concurrencia, la latencia mediana descendió de 153 ms a 26 ms, lo que equivale a una mejora mediana de 5,7 veces, observándose en la mayoría de las consultas mejoras de múltiples órdenes (hasta 12×). Bajo carga concurrente (10 solicitudes simultáneas), la latencia mediana se redujo de 157 ms a 38 ms, es decir, una mejora mediana de 3,9 veces, lo que indica que los beneficios se mantuvieron en gran medida bajo carga, aunque parcialmente atenuados por la contención. Se observaron mejoras de rendimiento en prácticamente todas las consultas evaluadas.

En conjunto, los resultados indican que el enfoque en memoria mejora de forma significativa la capacidad de respuesta en consultas acotadas a COMPOSITION y a EHR, si bien un subconjunto reducido de patrones de consulta puede requerir ajustes adicionales o la reversión al camino de ejecución convencional. En una consulta de laboratorio incluida en la evaluación, el tiempo de respuesta se redujo de aproximadamente 1,5–2,0 segundos a ~600 ms (sobre un subconjunto anonimizado equivalente al 10 % del entorno productivo). En términos generales, el AQL en memoria constituye un mecanismo de alto impacto para determinadas consultas, con evidencia inicial de mejora en latencia, aunque con limitaciones reconocidas que se describen a continuación. Los resultados muestran que el mayor beneficio se obtiene cuando el motor estándar de AQL o las consultas ejecutadas directamente en la base de datos no alcanzan un rendimiento adecuado, o cuando ciertas operaciones sobre COMPOSITION resultan difíciles de expresar de forma eficiente en SQL, especialmente cuando se requiere una combinación extensa de proyecciones y filtros. En estos escenarios, la base de datos subyacente suele presentar tiempos elevados debido a la complejidad estructural de openEHR.

Las mediciones de rendimiento evidenciaron mejoras incrementales tanto en el rendimiento global (throughput) como en la latencia en cola (tail latency). No obstante, dichas mejoras no se tradujeron de forma consistente en el cumplimiento de todos los objetivos de producción para algunas consultas. En las consultas centradas en el paciente, los resultados se mantuvieron estables independientemente del tamaño global del sistema, lo que sugiere que el rendimiento depende más del volumen de datos por historia clínica (EHR) que del tamaño total de la base de datos.

Cabe señalar que el uso de IMP modificó la distribución de carga desde la base de datos hacia el procesador AQL del HIP, reduciendo el consumo de recursos en la base de datos e incrementando el uso de recursos en el motor AQL de la HIP, incluyendo memoria RAM, CPU y ancho de banda de red.

## *4.2.Efectos de la indexación*

En términos generales, la estrategia de indexación demostró beneficios claros en rendimiento y escalabilidad, particularmente en consultas a nivel poblacional (consultas sobre el conjunto de pacientes). En pruebas de rendimiento a pequeña escala, la tendencia de ejecución de las consultas sin tablas índice aumentó de forma pronunciada con el crecimiento del conjunto de datos, mientras que las consultas que utilizaban tablas índice mostraron una tendencia aproximadamente logarítmica, lo que indica una mejor escalabilidad. En contraste, la sobrecarga en la ruta de escritura derivada del mantenimiento de las tablas índice durante las inserciones se mantuvo prácticamente constante a medida que aumentaba el volumen de datos.

En las pruebas realizadas sobre el gran conjunto de datos de ensayo (10 % de datos productivos anonimizados), las mayores mejoras relativas se observaron de manera consistente en los casos de uso transversales entre pacientes. Además, las tablas índice permitieron la ejecución de al menos una consulta que, en ausencia de indexación, excedía el tiempo máximo de ejecución (en particular, consultas transversales entre pacientes en un periodo de un año sin soporte de índices).

En las consultas centradas en el paciente, el factor de mejora fue menor, aunque igualmente relevante, dado que la consulta base ya presentaba un rendimiento comparativamente alto. Al igual que en el caso de IMP, los resultados para consultas centradas en el paciente se mantuvieron estables en distintos tamaños de sistema, lo que sugiere nuevamente que el rendimiento depende más del volumen de datos por historia clínica (EHR) que del tamaño total de la base de datos.

Los experimentos de mantenibilidad mostraron que la creación de nuevos índices es operativamente viable, aunque intensivo en recursos cuando se aplica a datos a escala productiva: iteraciones sucesivas reportaron tiempos de carga del orden de decenas de minutos a varias horas, dependiendo del volumen y del grado de paralelización. Los análisis de almacenamiento indicaron que las tablas índice son sustancialmente más pequeñas que la estructura genérica de datos de composiciones utilizada por el AQL estándar, lo que respalda la premisa de que la materialización de campos específicos reduce tanto el volumen de datos escaneados como la complejidad de las consultas. En el conjunto de prueba (117.196.656 composiciones; tamaño de almacenamiento aproximado de 1,38 TB), la tabla índice para resultados de laboratorio (consultas de un solo paciente) abarcó 8.227.707 composiciones y generó 188.840.434 filas, mientras que las acciones de vacunación (consultas transversales entre pacientes) abarcaron 13.708.414 composiciones y generaron 15.256.535 filas. El informe cuantificó además reducciones sustanciales de tamaño en comparación con los datos de composiciones (por ejemplo, los datos de composiciones eran aproximadamente 28,5 veces mayores que la tabla índice de resultados de laboratorio y aproximadamente 203 veces mayores que la tabla índice

de acciones de vacunación, en términos de tamaño de tabla), lo que refuerza que las tablas índice pueden aportar beneficios significativos en rendimiento manteniendo un coste de almacenamiento asumible para casos de uso específicos.

#### *4.3.AQL manual*

El SQL manual se ha utilizado en escenarios muy específicos en los que otras estrategias de optimización mostraron un éxito limitado. En particular, se empleó en consultas transversales entre pacientes que presentaban comportamientos problemáticos con superación del tiempo máximo de ejecución (timeouts). Si bien el SQL manual puede resultar útil para este tipo de consultas, nuestra experiencia indica que las consultas transversales orientadas a fines analíticos (investigación, epidemiología, estratificación de pacientes) se gestionan de forma más adecuada en repositorios OLAP independientes, que pueden sincronizarse mediante la funcionalidad de disparadores de eventos (event triggers) del HIP.

### **DISCUSIÓN**

La ejecución de un Repositorio Clínico de Datos (CDR) basado en openEHR a escala regional requiere abordar el rendimiento de AQL como un problema de optimización multicapa, que demanda soluciones especializadas en distintos niveles. En el despliegue realizado en Cataluña, las principales limitaciones de rendimiento derivaron del desajuste inherente entre (i) las características OLTP de los datos openEHR —centradas en el paciente, altamente normalizadas y versionadas— y (ii) la necesidad de dar soporte tanto a consultas interactivas por paciente como a consultas poblacionales transversales entre pacientes. Como consecuencia, diferentes estrategias de optimización resultaron efectivas para distintas clases de consultas, y las mejoras más consistentes se lograron seleccionando la estrategia más adecuada entre: patrones óptimos de diseño AQL, cambios en la ruta de ejecución (AQL en memoria), indexación a nivel de base de datos y ajustes específicos a nivel SQL para consultas almacenadas estables.

El AQL en memoria demostró ser un mecanismo de alto impacto para determinados patrones de consulta, especialmente aquellos en los que la traducción estándar de AQL a SQL genera planes de ejecución complejos o en los que las operaciones sobre estructuras openEHR (por ejemplo, múltiples proyecciones y filtros complejos) resultan difíciles de expresar de manera eficiente en SQL. En nuestras pruebas sobre un conjunto de datos similar al entorno productivo, el modo de ejecución en memoria redujo la latencia mediana de 153 ms a 26 ms bajo carga monohilo (mejora mediana de 5,7×) y de 157 ms a 38 ms con 10 solicitudes concurrentes (mejora mediana de 3,9×), lo que indica que los beneficios se mantienen en gran medida bajo carga. La indexación proporcionó mejoras medibles, aunque heterogéneas según la carga de trabajo, lo que confirma que añadir índices no constituye una solución universal para

el rendimiento de AQL. En consultas transversales entre pacientes (consultas poblacionales), las pruebas que combinaron nuevos índices y ajustes en AQL oscilaron desde comportamientos óptimos inferiores al segundo hasta respuestas de varios segundos en el peor caso. El rendimiento mostró una fuerte sensibilidad a la amplitud de la ventana temporal, siendo los intervalos más amplios consistentemente asociados a mayores tiempos de respuesta. Aunque los beneficios del AQL en memoria son significativos, todavía pueden ampliarse mediante optimizaciones actualmente en desarrollo, como el uso de mecanismos de caché para minimizar la recuperación masiva de datos desde la base de datos subyacente y la mejora de la trazabilidad.

Los resultados también pusieron de manifiesto que los patrones de diseño de consultas —especialmente la limitación de resultados y la paginación— pueden dominar el rendimiento en determinados escenarios: en un ejemplo, intervalos temporales amplios sin límites produjeron tiempos de respuesta de varios segundos, mientras que la aplicación de límites redujo el tiempo de ejecución a menos de un segundo.

Desde el punto de vista operativo, la propia creación de índices puede convertirse en un cuello de botella. Por esta razón, su implementación requiere alineación con la distribución real de los datos, estrategias de particionamiento y flujos de despliegue robustos que incluyan validación, mecanismos de reversión y ensayos en entornos similares a producción. Asimismo, el modelo de datos subyacente de la base de datos debe optimizarse específicamente para el rendimiento de AQL.

El enfoque más escalable a nivel de base de datos para cargas transversales entre pacientes fue la estrategia de tabla índice (enfoque integrado), que materializa rutas frecuentemente consultadas en estructuras relacionales y adapta la generación de SQL para dirigirse a dichas estructuras. En nuestra evaluación sobre datos similares a producción (>117 millones de composiciones), las tablas índice mostraron una tendencia estable a medida que aumentaba el tamaño del conjunto de datos: la ejecución sin indexación presentó un deterioro pronunciado, mientras que las consultas indexadas mantuvieron un comportamiento aproximadamente logarítmico con la escala. Este beneficio fue especialmente notable en consultas poblacionales transversales entre pacientes.

De manera relevante, la sobrecarga de almacenamiento no resultó prohibitiva: la evaluación reportó reducciones sustanciales de tamaño respecto a la representación genérica de datos de composiciones (por ejemplo, los datos de composiciones eran aproximadamente 28,5 veces mayores que la tabla índice de resultados de laboratorio y aproximadamente 203 veces mayores que la tabla índice de acciones de vacunación, en términos de tamaño de tabla). Esto respalda la premisa de que la materialización selectiva de campos reduce tanto el volumen de datos escaneados como la

complejidad del plan de ejecución. No obstante, las tablas índice introducen implicaciones de gobernanza y mantenibilidad, ya que son más adecuadas para familias de consultas estables y parametrizadas, y pueden derivar hacia una tabla índice por caso de uso si no se gestionan de manera sistemática.

Cuando ni la ejecución en memoria ni la indexación proporcionan un comportamiento aceptable, las consultas almacenadas con SQL manual constituyen una solución pragmática para cargas de trabajo estables y de alto valor: al especificar explícitamente el SQL y, en su caso, sugerencias al planificador para una consulta AQL conocida, los equipos pueden aprovechar la experiencia en bases de datos para estabilizar planes de ejecución y reducir la sobrecarga de planificación en consultas complejas. Este enfoque debe considerarse un recurso de último nivel, dado que requiere esfuerzo de ingeniería y conocimiento profundo para mantener alineadas la semántica AQL y el comportamiento SQL a medida que el sistema evoluciona.

Finalmente, la evidencia conjunta sugiere que, incluso aplicando la estrategia óptima según el tipo de consulta, algunos escenarios —especialmente agregaciones complejas transversales entre pacientes— pueden seguir siendo intrínsecamente difíciles de atender bajo garantías propias de sistemas OLTP (cumplimiento estricto de ACID, versionado y alta concurrencia de escritura) con latencias inferiores al segundo. En estos casos, resulta aconsejable una arquitectura híbrida: mantener el CDR openEHR como sistema de registro transaccional para flujos clínicos intensivos en escritura y trasladar las cargas analíticas a un repositorio optimizado para OLAP mediante exportaciones controladas. En este sentido, la capacidad del repositorio para activar exportaciones de datos ofrece un mecanismo para sincronizar sistemas externos optimizados para lectura sin comprometer el rol transaccional del CDR.

Aunque las estrategias presentadas se orientan a sistemas de gestión de bases de datos relacionales, podrían explorarse enfoques alternativos basados en otros paradigmas de bases de datos, como los documentales [6]. Investigaciones futuras podrían considerar una evaluación comparativa entre ambos enfoques.

### *5.1. Implicaciones prácticas*

La estrategia operativa que se desprende del trabajo realizado en Cataluña puede resumirse en: (i) realizar el modelado a partir de escenarios clínicos, identificando los puntos de datos más consultados y alineándolos con el diseño de las plantillas para facilitar la indexación; (ii) comenzar con directrices de diseño de consultas (limitación de resultados, paginación y patrones probados); (iii) aplicar AQL en memoria de forma selectiva para recuperaciones estructuralmente complejas centradas en un paciente; (iv) utilizar índices convencionales para campos del Modelo de Referencia (RM) de reutilización amplia; y (v) reservar el SQL manual para consultas almacenadas estables y críticas para el negocio.

## REFERENCIAS

[1] Piera-Jiménez J, Carot-Sans G, Ramiro-Pareta M, Nogueras MM, Folguera-Profítos J, Ródenas P, Jiménez-Rueda A, de Pando Navarro T, Mira Palacios JA, Fajardo JC, Ustrell Campillo J, Vela E, Monterde D, Valero-Bover D, Bonet T, Tarrasó-Urios G, Cantenys-Sabà R, Fabregat-Fabregat P, Gómez Oliveros B, Berdún J, Michelena X, Cano I, González-Colom R, Roca J, Solans O, Pontes C, Pérez-Sust P. A 25-Year Retrospective of Health IT Infrastructure Building: The Example of the Catalonia Region J Med Internet Res 2024;26:e58933 doi: 10.2196/58933

[2] Piera Jiménez J, Michelena Vegas X, Climent Fageda A, Valle Martín LI, Maymó Costa A, Dodas Perpinyà A, et al. Visió col·lectiva per al futur de la salut: bases per a un nou model assistencial i de sistemes d'informació. Barcelona: Departament de Salut; 2025. <https://salutweb.gencat.cat/ca/detalls/Articles/2025-07-10-llibre-blanc>

[3] Hertzum M, Ellingsen G, Cajander Å. Implementing Large-Scale Electronic Health Records: Experiences from implementations of Epic in Denmark and Finland. Int J Med Inform. 2022 Nov;167:104868. doi: 10.1016/j.ijmedinf.2022.104868. Epub 2022 Sep 14. PMID: 36194994.

[4] Torab-Miandoab A, Samad-Soltani T, Jodati A, Rezaei-Hachesu P. Interoperability of heterogeneous health information systems: a systematic literature review. BMC Medical Informatics and Decision Making. 2023;23:18. doi:10.1186/s12911-023-02115-5.

[5] Publication data - Plataforma de Serveis de Contractació Pública, Centre de Telecomunicacions i Tecnologies de la Informació de la Generalitat de Catalunya, 2023, <https://contractaciopublica.cat/en/detall-publicacio/944a111e-7bec-d408-0756-42a2549aa849/200041523>.

[6] Mateu Amengual F, Cox G, Mira i Palacios JA, Crossley J, Underwood J, Rodríguez G, Alonso C, Sanz de Acedo J, Pachot F. A document-first openEHR persistence layer for operational single-patient and cross-patient queries. BMC Proc. 2026. In press.