Canada Health Infoway
Inforoute Santé du Canada

HALO Working Group Meeting

Meeting Summary

Meeting Chair: Alex Reis			
Date and Time July 14, 2025,1:00pm – 2:30pm ET	**Location** Virtual	**Note Taker** Sadrina Petit, Senior Project Analyst, Digital Health Interoperability	**Next Meeting Date** July 21, 2025,1:00pm-2:30pm ET
Meeting Agenda: 1. Subscription Deep Dive – In Simplifier 2. Q&A / Collaboration			
Presenters			
• Alex Reis | Director, Digital Health Interoperability, Canada Health Infoway • Colin Kent-Shepherd | Software Architect, Hamilton Health Sciences			
Invited Guests			
Public			

---

1. **Welcome and Introductions**

   A. Reis welcomed all participants to the working group meeting and introduced Colin Kent-Shepherd. Meeting materials and recording of the session will be made available on the InfoCentral working group.

2. **Content Presentation**

   The Infoway team presented each of the agenda items as outlined above. The meeting focused on the HALO Pub/Sub model, using FHIR Subscriptions to push changes from the jurisdictional SOFA (where data is staged via $set-context and accessed with SMART tokens) back to Point of Care (POC) systems. We covered topic-based Subscriptions (FHIR R5 and the R5 to R4 backport), event number sequencing to ensure ordered processing, and security controls (auth headers, mTLS), with an option to use custom polling channels where needed.

   The presentation deck is available HALO Working Group Meeting

   The video recording is available HALO Working Group Meeting

3. **Questions raised during the working group meeting:**

   How are REST Hook notifications secured?

Include required auth headers (e.g., Authorization: Bearer <token>) in Subscription.channel.header[]; SOFA must echo them in outbound notifications. Jurisdictions may also require **mutual TLS (mTLS)** for additional assurance.

Can the same security approach be used for WebSockets?

Not the same headers. WebSocket channels use the **binding token flow** defined in the FHIR Subscriptions backport to authenticate and bind the socket to the subscription. Equivalent trust, different mechanism.

What's the difference between Subscription status = off vs "suspended"?

Base wording suggests off after too many errors/expiry; in practice HALO is using off as a pause / suspended state (no notifications delivered) *and* for error/expiry. Spec text will be clarified so implementers can rely on it as an intentional pause.

Is a polling model supported if I can't (or don't want to) receive pushes?

HALO v1 does **not** define a formal polling channel, but a workaround exists: create the subscription and **do not complete the handshake**. SOFA queues events; the client periodically calls **$events** to pull them. Behavior is implementation specific—jurisdictions should document details. A formal custom polling channel may be added if demand is strong.

If my EMR crashes or I miss notifications (e.g., during handshake), are events lost?

Events accumulate server side. On restart (or if handshake incomplete), call **$events** to retrieve any queued notifications you missed, then resume normal processing.

What do I do if I detect a gap in event numbers?

Use the **sequential eventNumber** in SubscriptionStatus to detect missing events. When a gap is found, call **Subscription/$events** (starting after your last processed eventNumber) to fetch and apply the missing events in order.

How do HALO implementations handle FHIR R4 vs R5? Do I need separate endpoints?

Everything *on the wire* in HALO remains **FHIR R4** (aligned to CA Core). HALO adopts the **R5 Subscriptions Backport** profiles/extensions to emulate topic-based behavior; you do **not** expose R5 endpoints. Internally, servers may map to R5 constructs if they wish, but clients speak R4.

Will R4 content ever be wrapped in R5 resources in notifications?

No. Notification Bundles carry **R4 resources** profiled per the Backport (e.g., Parameters used to represent SubscriptionStatus). Implementers consume R4; any R5 mapping is internal to the implementation.

Can HALO Subscriptions indicate which app/user/context triggered the event?

Not in the current release. Subscriptions are **system-level** (one per POC instance), so notifications don't identify the originating app/user. Options (e.g., Provenance, extended event metadata, message headers) are under consideration for a future version.

Do I have to support delete notifications? How do I distinguish between create/update/delete?

Current minimal logic: if a resource ID mapping exists → treat as update; if none → create. This is ambiguous for deletes. HALO is evaluating adding an explicit operation indicator (HTTP verb) or profiling guidance; jurisdictions may restrict delete permissions until clarified.