### Short Description
FHIRcast provides modern, simple, application context synchronization. We're working within the RIS/PACS community and beyond to standardize FHIRcast as the common, lightweight standard. We are continuing to test the more advanced features in proposed STU3.

Join this track to test with maturing implementations and new aspects of the specification.

### Long Description
Gain experience and collect feedback on the proposed STU3 specification, particularly on the small number of breaking changes, and the significant, optional addition of content sharing capabilities.

### Type
Test an implementation guide

### Submitting Work Group/Project/Accelerator/Affiliate/Implementer Group
InM

### Track Lead(s)
Isaac Vetter, isaac@epic.com

### Track Lead Email(s)
isaac@epic.com

### Related Tracks

### FHIR Version
R4

### Specification(s) this track uses
[https://build.fhir.org/ig/HL7/fhircast-docs/](https://build.fhir.org/ig/HL7/fhircast-docs/)

### Artifacts of focus
[https://build.fhir.org/ig/HL7/fhircast-docs/4-6-content-sharing.html](https://build.fhir.org/ig/HL7/fhircast-docs/4-6-content-sharing.html)

### Expected participants
We usually have ~6 or so implementing/testing developers.

### Zulip stream
[https://chat.fhir.org/#narrow/stream/179271-FHIRcast](https://chat.fhir.org/#narrow/stream/179271-FHIRcast)

### Track Kick Off Call
Track Details

Checkout our reference implementation and other opensource FHIRcast projects at: https://github.com/fhircast.

Hub

The Hub manages users' sessions, accepts subscriptions, is responsible for notifications and is closely aligned with the SMART server and FHIR server.

- Must accept new subscriptions.
- Must verify subscribed callback urls.
- Must accept context change requests from subscribers.
- Should support websockets as described in the spec.
- May POST notifications to subscribed callback urls.

Subscriber

Subscribes to a user's session topic on the hub, receives and processes event notifications.

- Must request subscription to topic and events on hub url.
- Must accept event notification.
- May request context changes.
- Should support websockets as described in the spec.

Basic Scenarios

Subscriber creates new subscription; hub respond with 202

:Action: Subscriber POSTs WebSub subscription request to hub for a specific hub.topic and hub.events.
:Precondition: Subscriber must know hub base url, hub.topic (session identifier) and supported events. Use "Patient.open", "Patient.close" for events for Connectathon.
:Success Criteria: Hub persists subscription.
:Bonus point: Subscribe via websocket.

Subscriber is notified of user session change per subscription

:Action: Hub/Publisher POSTs notification to subscriber for a subscribed event and topic/session.
:Precondition: Subscriber subscribes correctly, see previous scenario.
:Success Criteria: Subscriber responds with 2xx code.
:Bonus point: Notify client via established websocket connection.

Subscriber requests user session change

:Action: Subscriber POSTs context change request to Hub's topic url.
:Precondition: Subscriber subscribes correctly, see initial scenario.
:Success Criteria: Hub responds with 2xx code.
:Bonus point: Hub accepts context change and broadcasts corresponding notification.
:Bonus point: Hub rejects context change and subscriber handles synchronization failure.

Subscriber is notified of topic/session id, hub url during SMART app EHR launch from Hub

:Action: Subscribing client launches from Hub, according to SMART EHR launch.
:Precondition: n/a
:Success Criteria: Subscriber requests fhircast scope and receives hub base url and hub.topic in SMART launch and performs scenario #1.

Subscriber is notified of topic/session id, hub url during SMART app standalone launch

:Action: Subscribing client initiates SMART standalone launch using Hub's OAuth2 server, according to SMART EHR launch.
:Precondition: n/a
:Success Criteria: Subscriber requests fhircast scope and receives hub base url and hub.topic in SMART launch and performs scenario #1.
:Bonus point: Subscriber evaluates multiple returned sessions and chooses between them.

Advanced Scenario

Implement "content sharing" as described in this draft specification.