HEART Use Case: Elderly Mom with Family Caregiver

December 12, 2015 - Updated Sequence Diagram with link via http://bit.ly/heart-mom
November 17, 2015 - Added a sequence diagram based on the Baseline HEART profile doc .

Setup

Formatting (copied from OAuth use-case here)

Where this use case reflects a choice intended to inform the HEART WG's profiling deliverables that may vary against use cases that reflect other choices, the notation **[CHOICE: *description*]** appears. This choice should appear in the title of the use case in brackets to help distinguish it from other close variants.

Where this use case reflects a discussion point for the HEART WG's profiling efforts, the notation **[PROFILING]** appears.

Where this use case contains detail that is believed to be peripheral to the HEART WG's profiling deliverables, the notation **[PERIPHERAL]** appears. The point of this detail is to give real-life "color" to the use case.

The Problem

I'm the Custodian, my 89 y/o mother Alice's healthcare proxy, and I live in a different state. My mother is healthy and has a live-in caregiver that helps her schedule and get to health related services. My problem is to keep my mother away from the side-effects of overly aggressive treatments that are constantly being offered to her. This is "shared decision making" but, for the most part, I'm the party sharing the decision-making with various health services providers.

Monitoring my mother's encounters with the healthcare system from 200 miles away is a constant struggle. My immediate goal is to be notified of every interaction with the health care system. A simple notification service and convenient, linked access to my mother's Medicare claims interface and the provider's' electronic health records (EHR) interface would be a huge time saver, improve patient engagement, and reduce the risk of unwarranted procedures and costs.

Ecosystem parties

- **Alice**: an individual; a patient who consumes healthcare services and participates in shared decision making regarding her care. Alice is the underlying resource owner (principal) but she has allowed the Custodian the ability to manage her access policies on her resource server. The means by which Alice delegates permission to her Custodian to further delegate permissions is out of scope for this use case. [PERIPHERAL]
- **Custodian**: an individual. Alice's family caregiver and healthcare legal proxy.[PERIPHERAL] For this use case is the Resource owner (Authorized to control resource). The custodian never impersonates Alice. The means by which Alice delegates permission to her Custodian to further delegate permissions is out of scope for this use case. Assumption: Custodian has their own digital identity and services can differentiate between Alice and Custodian access to information. [PERIPHERAL]
- **Primary Care Provider** (PCP): a health care professional who will see Alice on a regular basis for common medical concerns; end user of an electronic health record (EHR) system, an enterprise cloud-based information system which tracks many patients' medical information.
- **Payer** (Medicare) system operator: a provider of a payment for claims by the PCP and others providing covered services to Alice, a cloud-based information system which tracks Alice's medical information for her and her healthcare proxy where the healthcare proxy is the end user with authority over her data where the Payer supports many such end users.

Technical preconditions

- The EHR system and the Payer system both use the standard FHIR*[1], API as their interface for access to Alice-specific resources.
- The EHR system and the Payer system both use UMA to protect their FHIR APIs. (Both allow dynamic registration of RS and client.)
- The EHR and Payer systems are not part of the same federation. Alice and FHIR are their only common denominator.
- For Alice's registration (in person) at the EHR system:
    - Alice has a simple, globally-routable identifier, such as an email address [PERIPHERAL] (or some other "out-of-band" electronic communications channel through which the PCP can send a verification request).
    - Alice carries a smartphone [PERIPHERAL] (or some other mobile device that enables her or her Custodian to fulfill the verification request during her in-person visit to the PCP).
    - Alice is able to use a PCP-provided kiosk in person.
- Alice has an existing account with a Payer and login credentials to access it.
    - She has provisioned it with basic demographic data, bank, and insurance information. [PERIPHERAL] (or some similar list of personal attributes).
- Alice and her Custodian do not necessarily have access to the data themselves, but they do have the ability to control access to the data via an authorization server. [PERIPHERAL]
- Alice and her Custodian have different digital identities and accounts. [PERIPHERAL]
- The Custodian and Alice benefit from registering indirectly through the AS as opposed to having to establish a separate delegation relationship with each RS.

OAuth entity roles

- **Protected resource** (PR): Online information or API that is access controlled through OAuth. Note that APIs can allow both "consumption of data" (read operations) and "insertion of data" (write operations) by authorized entities.
- **Resource owner** (RO): An entity that has OAuth access control rights to an online resource. The RO may not, however, have other "ownership" rights, such as the right to change data values within that resource.
- **Authorization server** (AS): An entity that issues OAuth access tokens representing the client's authorization for access on behalf of the RO.
- Resource server (RS): An entity where the PR resides. In OAuth, the AS and RS are typically "tightly coupled" and run by the same organization (by contrast, in UMA, entities with these names might not be).
- **Client**: A web or mobile application (or even an IoT device) used by the RO that seeks and gains access tokens from the AS in order to access the PR. Access may be limited (scoped) to a subset of possible API operations. The RO can typically visit the AS anytime to revoke the token.

UMA entity roles

- Protected resource (PR): Same as OAuth.
- Resource owner (RO): Same as OAuth; the “user” in User-Managed Access.
- Requesting party (RqP): An entity that seeks access to a PR. May or may not be the same party as the RO.
- Requesting party token (RPT): An UMA access token.
- Authorization server (AS): An entity that issues RPTs representing the authorization of the client and the RqP operating it for access.
- Resource server (RS): Same as OAuth. In UMA, the AS and RS can be “loosely coupled” and run by different organizations or entities, enabling the centralization of multi-RS management, fine-grained authorization modification, and RO choice of AS (by contrast, In OAuth, entities with these names are typically tightly coupled).
- Client: A web or mobile application (or even an IoT device) used by the RqP that seeks and gains RPTs from the AS in order to access the PR. Access may be limited (scoped) to a subset of possible resource sets and API operations on them.

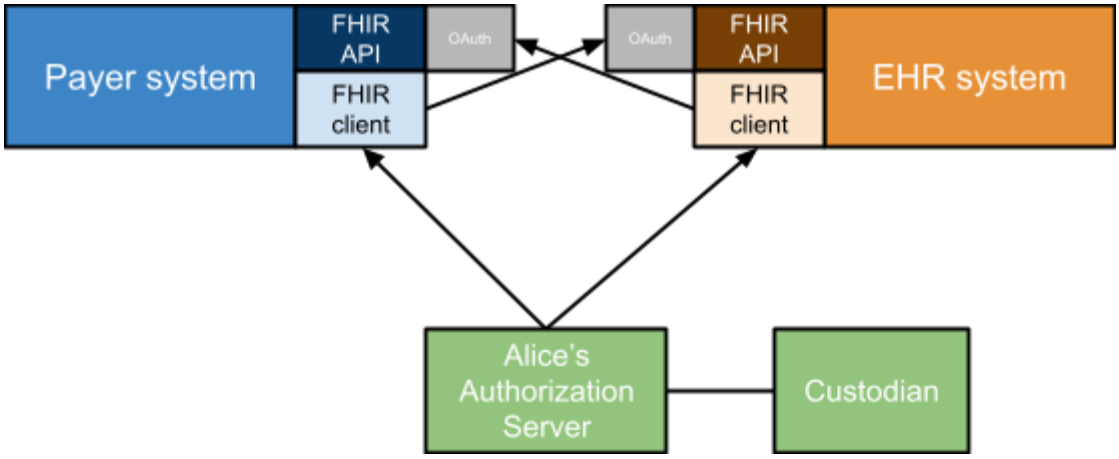
Entity definitions

- **Protected Resource** (PR): Online information or API that is access controlled through UMA. Note that APIs can allow both “consumption of data” and “insertion of data” by authorized entities.
- **Resource Owner** (RO): An entity that has OAuth access control rights to an online resource. The RO may not, however, have other “ownership” rights, such as the right to change data values within that resource.
- **Authorization Server** (AS): An entity that issues OAuth access tokens representing the RO’s authorization for access.
- **Resource Server** (RS): An entity where the PR resides. In OAuth, the AS and RS are typically “tightly coupled” and run by the same organization (by contrast, in UMA, entities with these names might not be).
- **Client**: A web or mobile application (or even an IoT device) used ***by anyone*** that seeks and gains access tokens from the AS in order to access the PR. Access may be limited (scoped) to a subset of possible API operations. The RO can typically visit the AS anytime to revoke the token.

Party-to-entity mappings

- **EHR system**: The OAuth RS for Alice’s and others protected health information (PHI) and, reciprocally, the client for the Payer system.
- **Payer system**: The OAuth RS for Alice’s and others’ protected personal health records and, reciprocally, the client for the EHR system.
- **Alice**: the RO at both the EHR system and the PHR system for her own PRs.

- **Custodian**: Alice's healthcare proxy with access to all of the portals and interfaces that Alice has.



Use Case Steps

Step 1. Initial Patient-PCP Contact: Not in Person (copied from OAuth use case)

Alice calls a PCP's office over the phone [PERIPHERAL] (or another person makes an appointment for her, or she contacts the office in another fashion) to begin enrollment and book her first appointment. The PCP's office creates a temporary [PERIPHERAL] patient record for Alice in the EHR system and schedules an appointment for her. This patient record is a PR.

Step 2. Initial Patient Visit to PCP's Office: Practice Registration and Portal Enrollment

Alice arrives for her scheduled appointment and registers at the front desk. The following series of actions take place:

1. She is identity-proofed using her driver's license and Medicare card [PERIPHERAL] (or by some other method -- proofing to some defined "level of assurance" is a governance decision but the method is peripheral), which are scanned. The images are stored in the PCP office's EHR system. As a result of this proofing, Alice's record is now marked as "known to the practice." This is the registration process.
2. She tells the PCP's registration desk which Authorization Server system she uses [PERIPHERAL] (or enables binding of her EHR record to her Payer record in some other fashion -- binding is essential but the method is peripheral). The doctor's office registration process includes sending a verification email to Alice's Custodian through

her Authorization Server. This begins the enrollment process and includes the binding of Alice's identity to the record.

3. The *Custodian* completes the email ***verification*** using her smartphone asynchronously - the PCP need not wait. [PERIPHERAL] (or by some other method -, the Custodian can now log in to view Alice's protected resources in the PCP's EHR at any time). Notice that verification and enrollment could be completed remotely.
4. While at the PCP's office, Alice or the Custodian is asked to log in to her newly provisioned EHR account and authorize a linkage of her Payer account [PROFILING] (some choice of flow), by the introduction of the EHR system to her Payer system so that the former can become an OAuth client of the latter, exchanging personal data with it as long as she doesn't revoke its access token.
5. Somewhere during or after the process of enabling the PCP's EHR system to become a client for her Payer system, she is given the opportunity by her Authorization Server system -- which she has just logged into for consenting purposes -- to make the reciprocal authorization possible, that is, to enable her Payer system to become a client in turn to her PCP's EHR system [PROFILING]. The two health record systems can now fully commence exchanging her personal data in an automated yet consented fashion, according to the scopes the client on each side was granted, where each client might be able to both "consume" data for which the server on the other side is authoritative, and "inject" data for which it itself is authoritative.

Step 3. Initial Patient Visit to PCP's Office: Examination Room

1. Alice is taken to the examination room. Her PCP conducts a physical examination and records the clinical findings in the EHR system.
2. PCP prescribes a medication (possibly informed by the Payer's formulary) which triggers a transaction with the Payer such as a benefits check.
3. The Custodian receives notice from Alice's Authorization Server that a medication order has been placed and, later, from Alice's Payer that a claim has been filed by the PCP with a link to a secured service to view the details. [PERIPHERAL] If the notice of activity comes from the Authorization Server, the Custodian would benefit from federated single sign-on to the linked protected service at the PCP and/or at the Payer.

Step 4. End of Patient Visit: Back Home

Alice returns home. Her Custodian receives email notifications from Alice's Authorization Server [PERIPHERAL] (or text messages or some other "pushed" communications based on her provided contact information and consent to use it) that information has been updated in both her Payer system and her PCP's EHR system. *[Note: the authorization server does not know every time the access tokens it issues are used. There would need to be a notification hook made available during the authorization process, most likely, but future work is needed here. Much of this is being worked on at the policy level in the UMA Legal working group.]*

Discussion

This use case highlights the following facets of HEART's charter to serve individual-centric, privacy- and security-sensitive RESTful health data sharing, leveraging a mix of technologies variously including OAuth, OpenID Connect, UMA, and the FHIR API:

- Technology to implement the OAuth security protection of RESTful APIs as described in this use case exists today, and is widely implemented and understood. The person-centric approach in this use-case allows us to separate out and hold off on profiling scopes specific to healthcare until after we discuss the "pure" authentication and authorization scopes that would apply to any vertical and API. For example, these are: authentication methods and profiles, client certificates, period of validity, read / write permissions, notification endpoints, cancellation terms.
- The FHIR API is very new, but its RESTful nature has supported rapid experimentation and piloting.
- Authorization Servers (or for that matter PHRs or HIEs) such as the one imagined to be available for Alice to choose freely, and interoperate with providers or payers on a FHIR basis, are not widely available on the market at this time.
- UMA improves Alice's and her Custodian's experience by giving them a primary point of access across multiple service providers. **This solves the Alice-to-Alice N problem.** OAuth alone gives Alice's Custodian the ability to consent to, and revoke, application access to APIs, but is not individual-centric in that the security relationships it forges are pairwise.
- By sharing responsibility with a patient-controlled Authorization Server, the Resource Server substantially avoids responsibility for delegated access and improves patient engagement. **This solves the Alice-to-Custodian delegation problem**.
- By allowing the direct transfer of information between the EHR and the Payer, UMA substantially reduces the provenance problem introduced by Personal Health Records and some health information exchanges as intermediaries. **This solves the Alice-to-Bob Directed problem**.
- This use-case does not require either the EHR or the Payer to federate or otherwise agree on a common sharing authorization form. Each of the institutional actors can present the patient signed into their portal their own content and design for their **Release of Information (ROI) Form** just like they do in the paper world. The lookup and capture of Alice's Authorization Server endpoint URI as an additional field on the ROI Form can be automated using well-known standards such as WebFinger or OpenID Connect.
- This use-case does not have a directory / discovery component. However, if the Resource Server chooses, they can add a field to the ROI Form's Resource Owner block that documents a patient's discoverable persona or identifiers. This would mitigate the common problem of "opt-in" to a health information exchange directory and solve the **Alice-to-Bob HIE problem**.

- By introducing a patient-specified Authorization Server, the burden and breach risk of patient notification shifts away from the Resource Server institution.
- Because it contacts a patient-specific server for every protected resource, UMA can significantly reduce the Resource Server's risk of a breach and the cost of a breach. The real-time accounting for disclosures provided by the Authorization Server reduces the time that breaches are discovered to minutes rather than months.
- Because it introduces a different patient-specific resource for every protected resource, UMA can significantly reduce the Resource Server's risk of a large-scale breach.
- An "online and mobile-friendly Alice" is a simplifying assumption for a great many benefits that can accrue to all of the parties in the ecosystem, including data accuracy, automation of consent, and PCP office efficiency. Alice or her Custodian does account linking as needed, typically in real-time.
- Patient mediated/centered exchange - as this use case demonstrates - avoids the patient match problem and minimizes the patient identity issue.

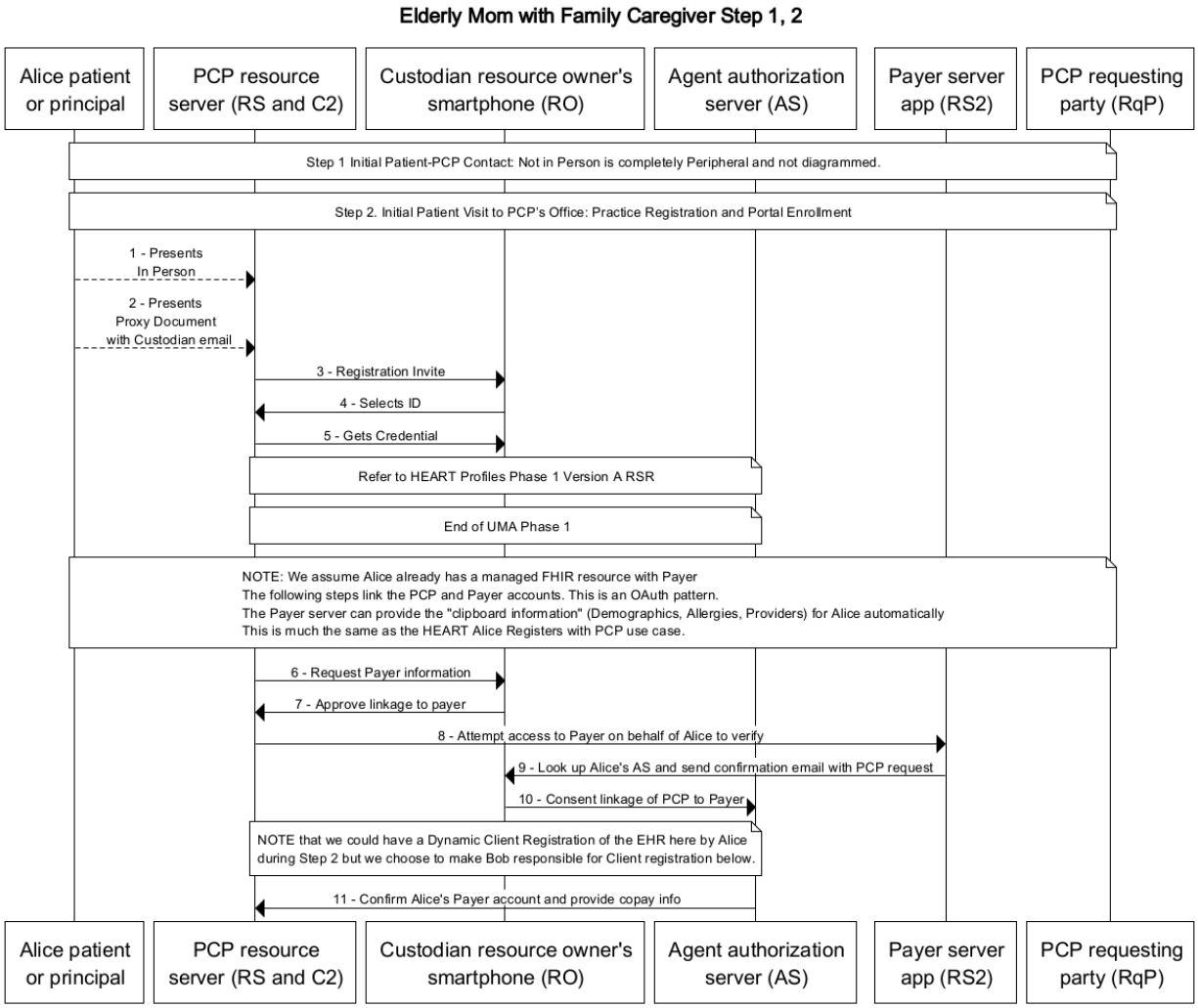
Glossary

*1 - FHIR stands for Fast Healthcare Interoperability Resources. See, e.g.,http://www.hl7.org/implement/standards/fhir/summary.html orhttp://www.hl7standards.com/blog/2013/03/26/hl7-fhir/ orhttp://wiki.hl7.org/index.php?title=FHIR

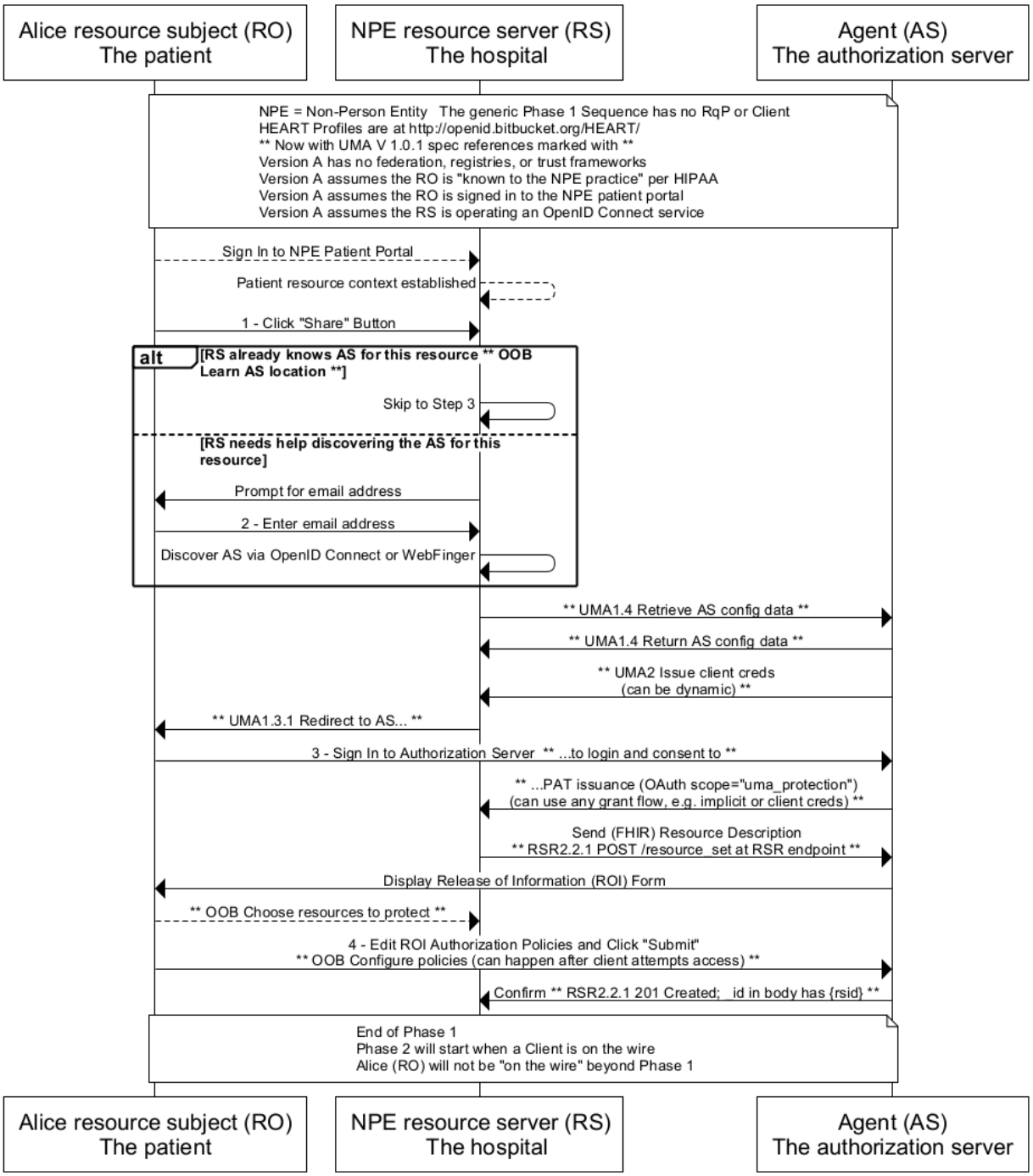
HIMSS Delegation Use case (URL)

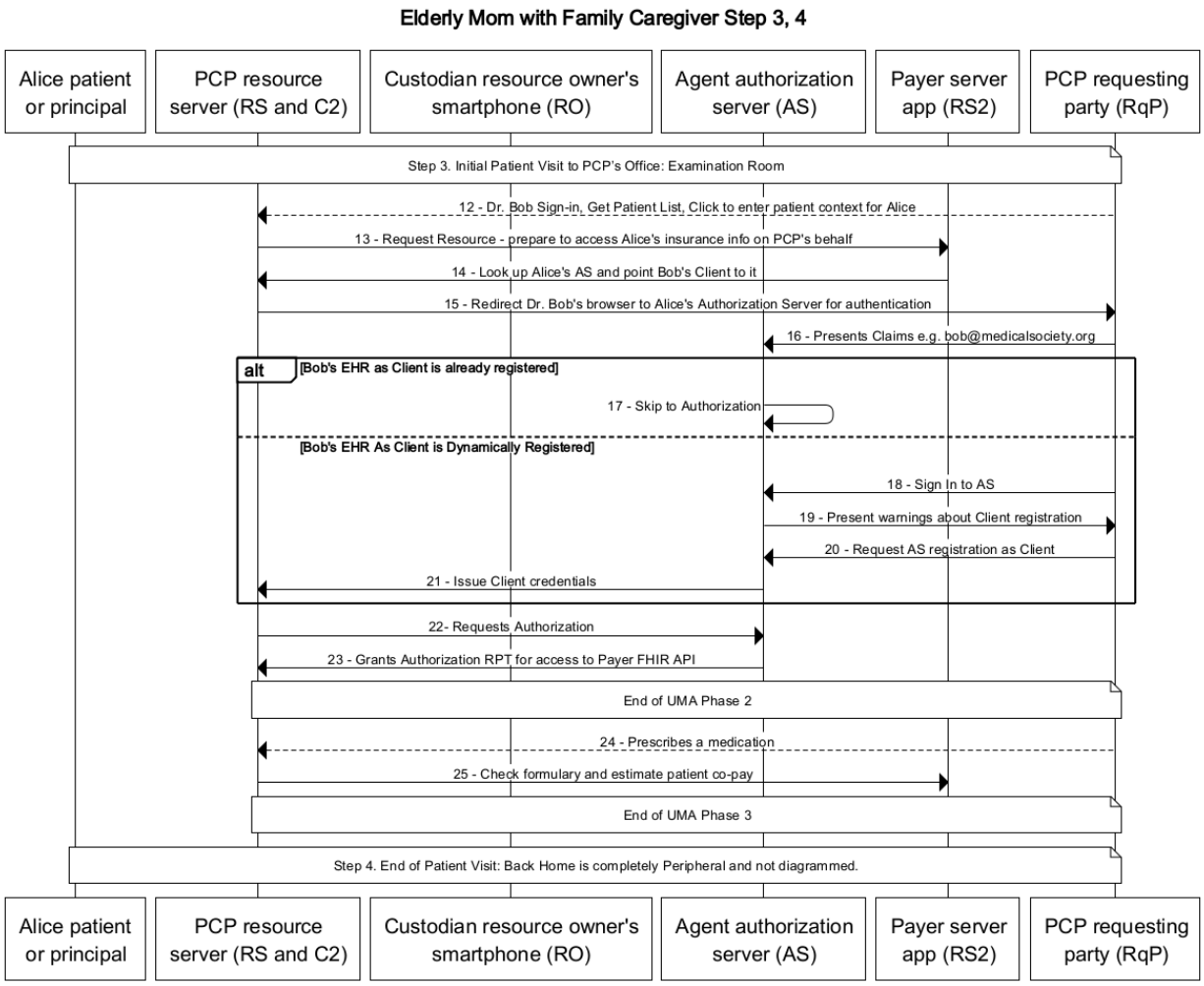
Sequence Diagrams

December 12 Sequence Diagrams



HEART Profiles Phase 1 Version A RSR





Nov 17 Diagram is obsolete

Here and source below. A Baseline HEART description doc is ay http://bit.ly/HEART-seq

```
title Elderly Mom with Family Caregiver Sequence Diagram

participant "Alice patient\nor principal" as RP
participant "EHR resource\nserver (RS)" as RS
participant "Custodian resource owner's\nsmartphone (RO)" as RO
participant "Agent authorization\nserver (AS)" as AS
participant "Payer client\napp (C)" as C
participant "Bob requesting\nparty (RqP)" as RqP
note over RP, RO, RS, AS, C, RqP
 Numbers correspond to description at http://bit.ly/HEART-seq
end note


RP->RS: 1 - Presents\nIn Person
RP->RS: 1a - Presents\nProxy Document\n with Custodian email
RS->RO: 1b - Registration Invite
RO->RS: 1c - Selects ID
```

```
RS->RO: 2 - Gets Credential
RO->RS: 3 - Sign In to EHR Portal
RS->RO: 4 - Display PCP ROI Form
RO->RS: 5 - Specify Auth'z Server (AS)
RO->RS: 6 - Submit Resource Description
RO->AS: 7 - Sign In to Agent Portal

RS->AS: 8 - FHIR Resource Description
AS->RO: 9 - Display Resource Policies
RO->AS: 10 - Confirm Authorization Policies
AS->RS: 11 - Confirm\nResource Registration
RS->AS: 12 - Consent Receipt

note over RO, RS, AS
 End of UMA Phase 1
end note

note over RS, AS, C, RqP
 - Patient already has a managed FHIR resource with Payer (or PHR).\n- PCP wants access to
Payer FHIR resources, asks RO for a Payer link, via SMS or email.\n- Custodian signs in to
Payer, presents RS URI. This is the patient match. HIE RLS would be an option.\n
- Custodian (OAuth) or policy (UMA) will now link PCP and Payer (PHR) in both directions.\n-
Note that both the RqP registration and the C registration steps are relative to the AS and
optional
end note

C->RS: 13 - Request Resource
RqP->AS: 14 - Presents Claims\ne.g. bob@medicalsociety.org
AS->RqP: 15 - Gets Credential
RqP->AS: 16 - Sign In to AS
RqP->AS: 17 - May need to Register Client
AS->C: 18 - Consent Receipt
C->AS: 19 - Requests Authorization
AS->C: 20 - Grants Authorization RPT

note over RS, AS, C, RqP
 End of UMA Phase 2
end note

C->RS: 21 - Access FHIR Resource using RPT
RS->AS: 22 - Accounting for Disclosure

note over RS, AS, C, RqP
 End of UMA Phase 3
end note
```