How UMA Contributes to Solving the IDESG Healthcare Relationship Location Service Use Case

@UMAWG

tinyurl.com/umawg

19 Jan 2014

kantara INITIATIVE™

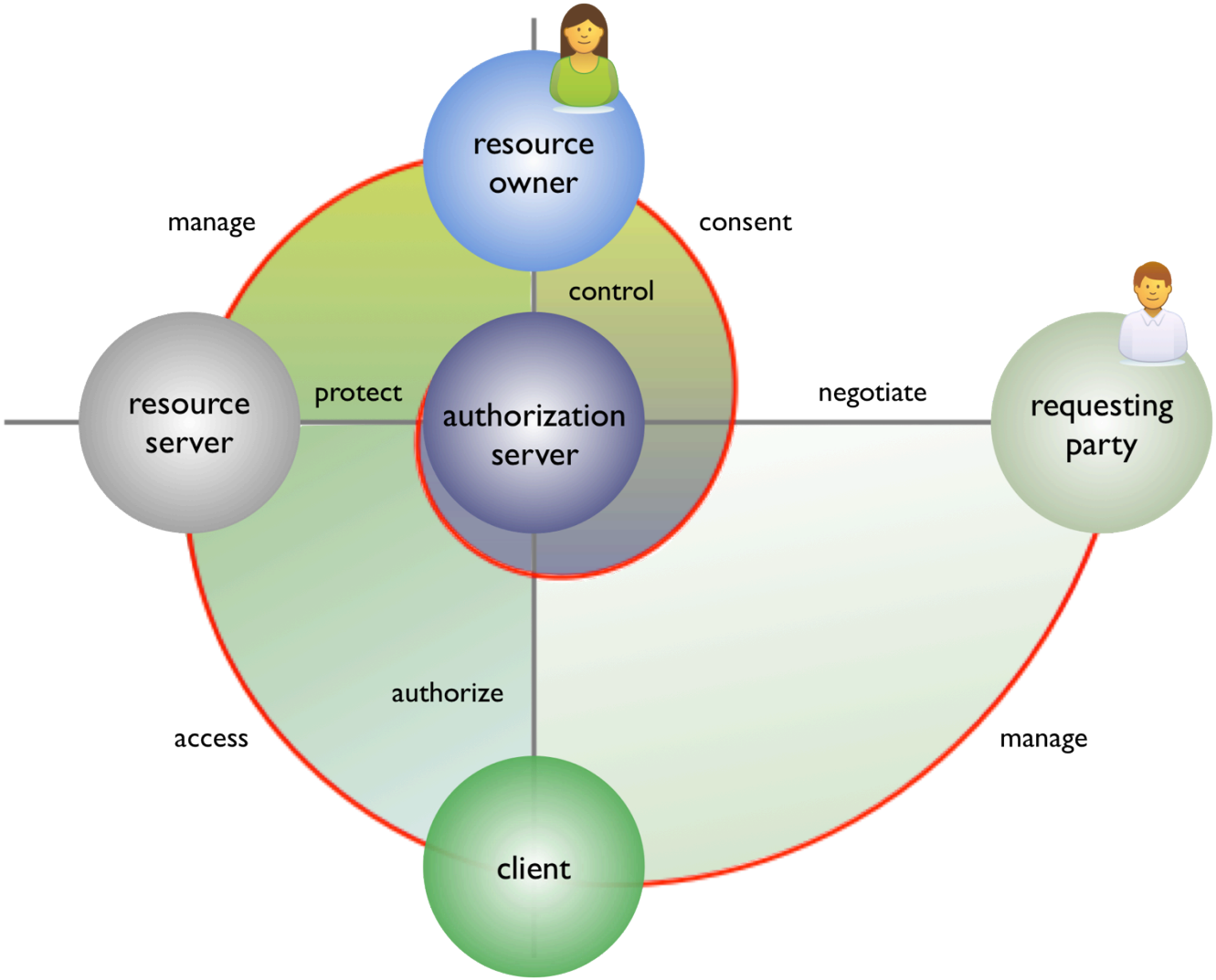
Relevant links

- IDESG use case:
  https://www.idecosystem.org/wiki/Health_IT_Record_Location_Service_(Data_Aggregation)
- UMA use case analysis working document:
  https://docs.google.com/document/d/1WS4c2bxAvHiDXFrWLrpCCRlvTYwmSTyV8C0fFj9VIOM/edit?usp=sharing

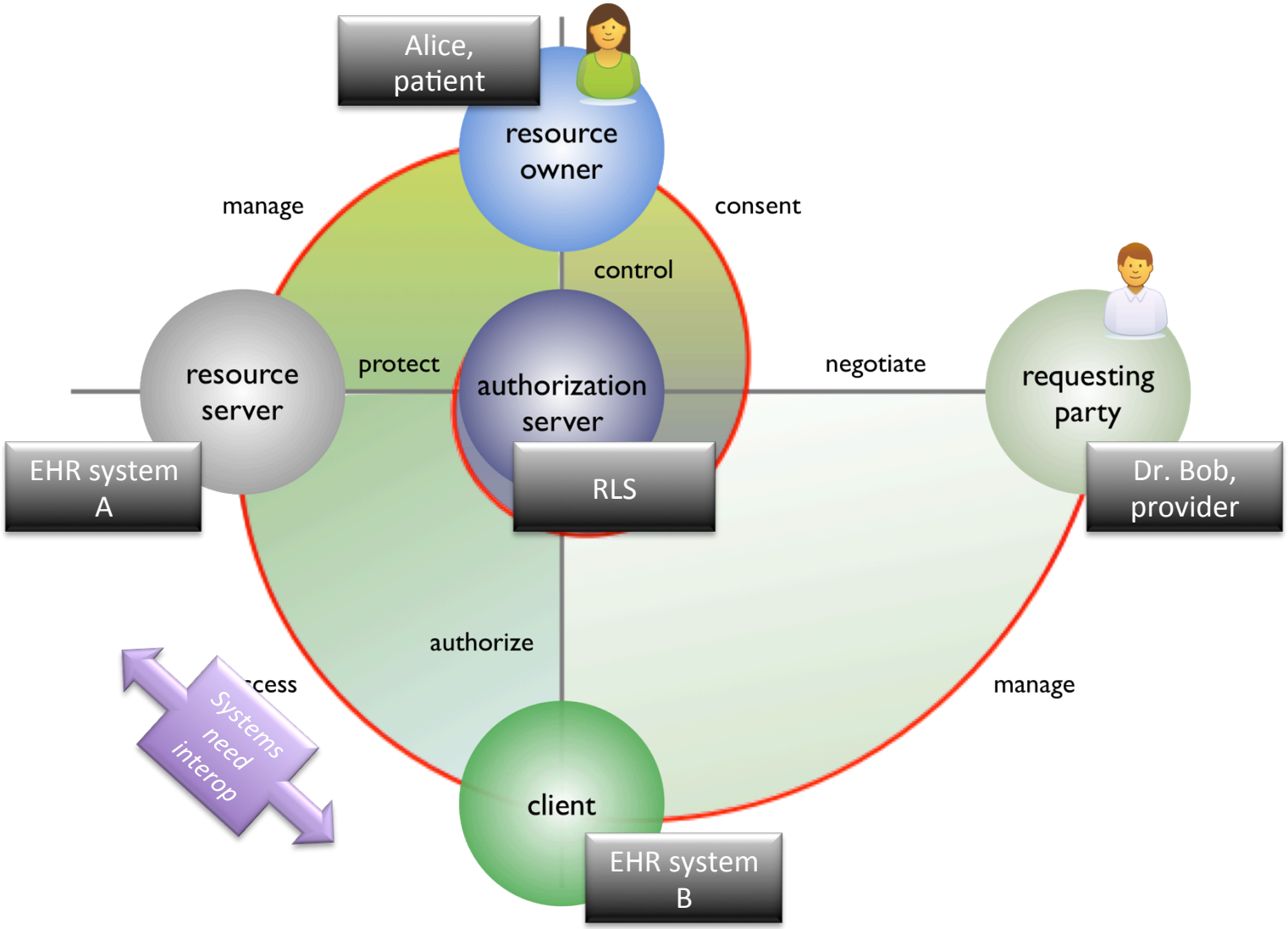
Assumptions

- Today the emphasis is on data aggregation; in future it will switch to *controlled access to distributed data* instead
- Patients in question will have an online presence (e.g., can log in to patient portals etc.) in future
- Even in cases where patients can't *control* sharing of their data by others, they must be able to *monitor* it

UMA actors


resource owner
manage
consent
control
resource server
protect
authorization server
negotiate
requesting party
authorize
access
manage
client

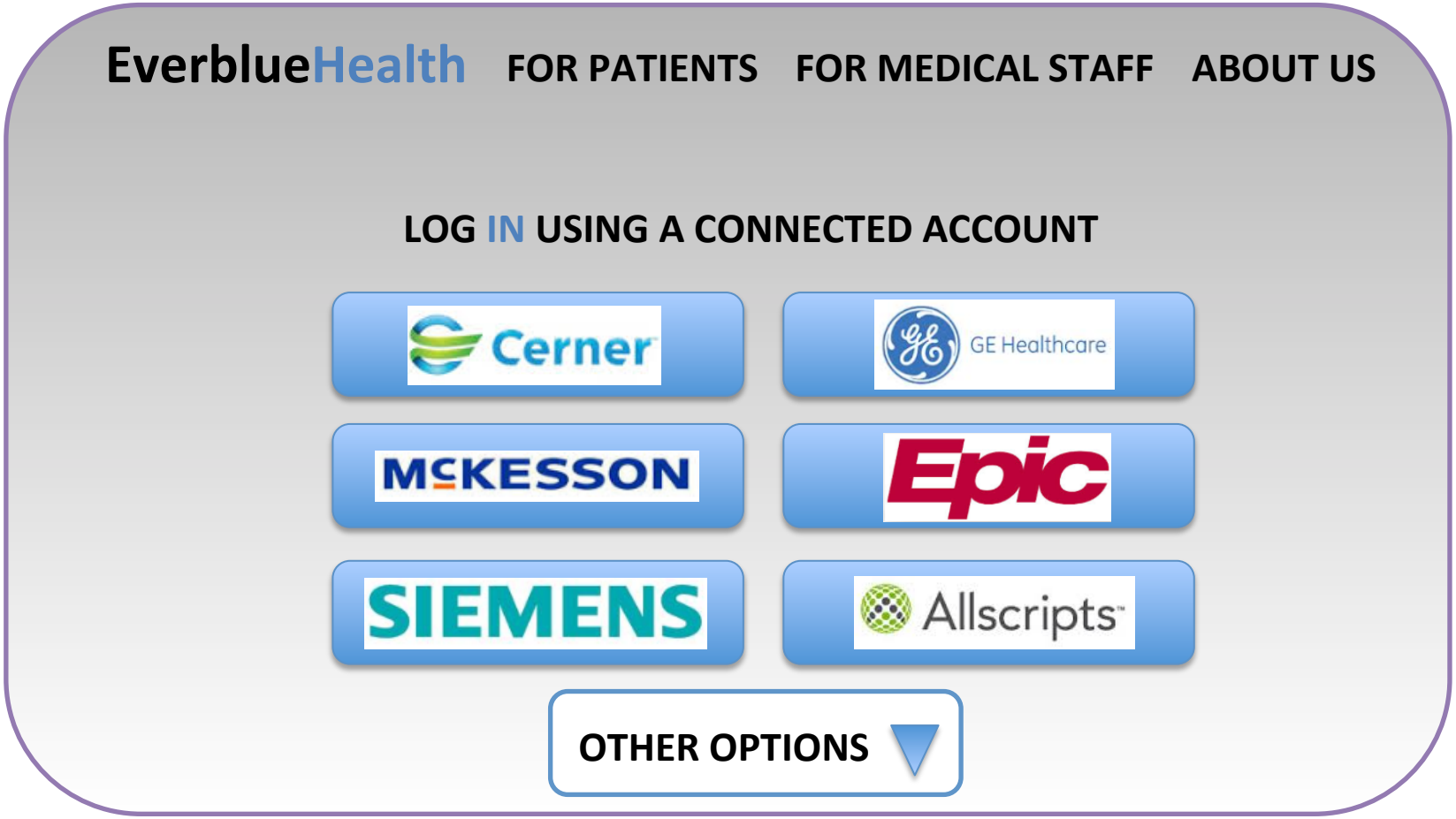
Mapping to use case actors



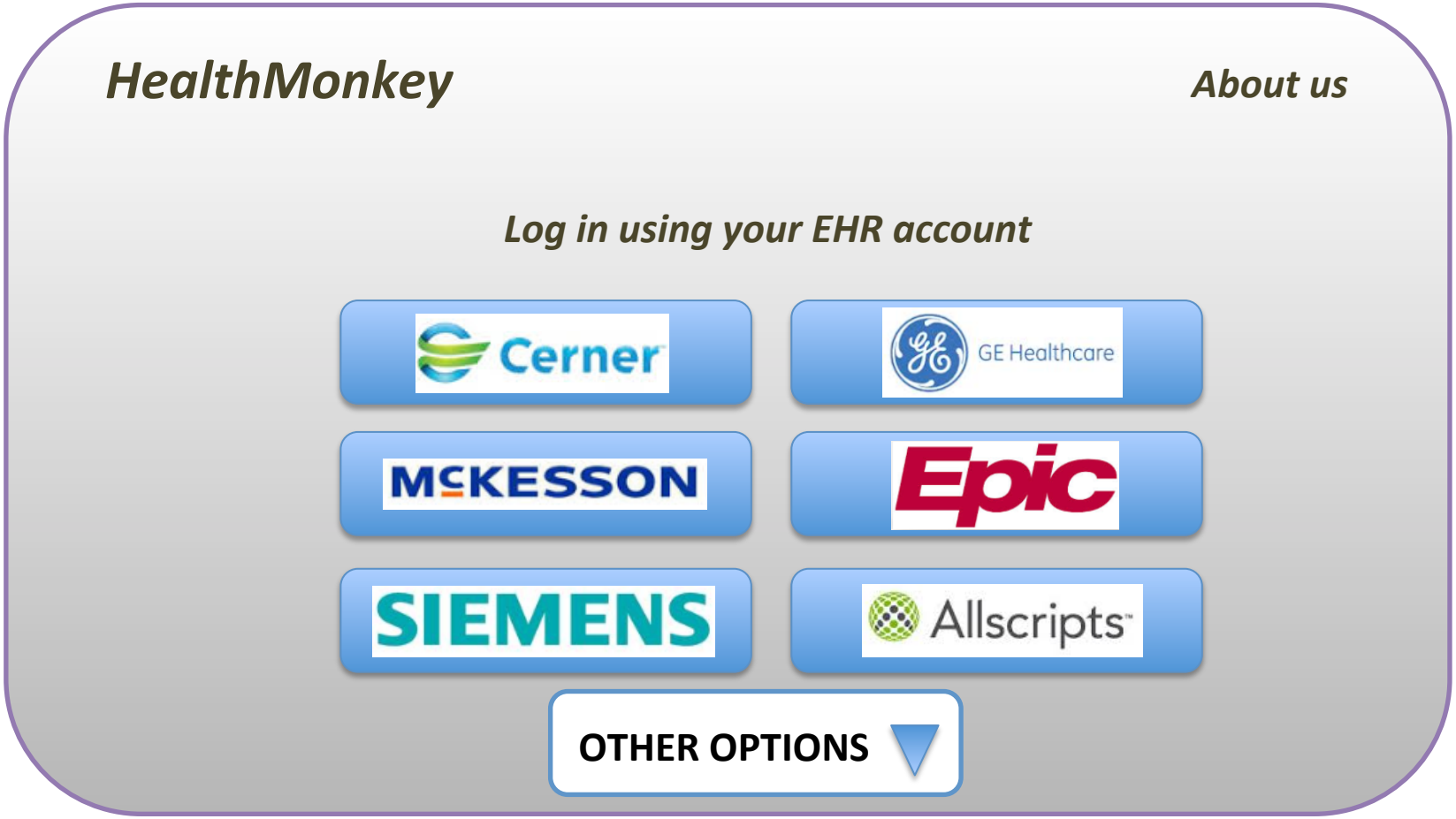
Alice needs to log in to EHR-A (RS) and RLS (AS) – how ?

- UMA is agnostic, but this matters to RLS functioning
- Some obvious options:
  - Log in to each with a local account
  - Log in to one or both with a federated account from elsewhere
    - Social and unmanaged ("unverified")
    - Proofed, secure, and nonrepudiable ("accountable")
  - Log in to one (as relying party [RP]) with an account from the other (as identity provider [IdP])
- Let's assume federation on an "accountable" basis through EHR vendors

Patient portal at provider



Patient data-sharing control console at RLS (could be run by insurer)



General discussion points

- Data is distributed, but its control is centralized
  - Gives some privacy-enhancing properties
- Only resource servers introduced to the hub are within Alice's "monitoring and control sphere"
  - Any data sources outside the system can't be tracked
- Trust frameworks/participation agreements still have a big role to play in governing sharing
  - But with Alice able to participate more fully than before
  - And with calculable rights based on UMA's Binding Obligations

Analyzing the use case assumptions

1. Backend system information is out of scope.
2. All touch points between RLS and Providers available via accessible APIs
3. There is an existing Participation and/or Federation agreement between Provider and RLS
4. RLS may support a Master Person Index (MPI) with one or many personas for each identity contained within the MPI
5. RLS provides optional identity audit service so patient can manage relationships reported for different personas

- True for UMA because it's about interfaces only.
- True for UMA because it standardizes protection and even enables app-specific standardization
- UMA anticipates "access federation" agreements on top of its Binding Obligations
- UMA is silent on this but has a channel to support profiling the AS and RS doing this at a level "above" UMA
- UMA enables accounting of disclosures but would require further sector-specific profiling

Analyzing the use case requirements

1. Patient consent for Provider to send relationship information to RLS
2. Patient portal or other means for patient to audit and submit corrected information in the RLS
3. Digital, real-time fulfillment of HIPAA Accounting of Disclosures and related public disclosure laws

- UMA **achieves this** through hub/data source introduction, protection API token (PAT) issuance, and Binding Obligations
- If the hub is also a data client, patient/resource owner **can do this** in a trackable way
- UMA **enables** accounting of disclosures but would require further sector-specific **profiling**

Next steps

- Build and validate "UMA++" swimlane flows against process flows in use case