

# Exercise 9: Semantic Hypermedia (Hypermedia I)

## Group 1

### Editorial Notes:

- For Task 2 we added an ADR explaining why we want to have a separate service for the Crawler. We then changed the text in Task 2 to discuss the architectural impact of the auction house discovery

### Task 1: Semantic Hypermedia Overlays

#### Critical reflection on collective decisions for the design of the semantic hypermedia overlays:

- One of the most contended decisions made by the interoperability group was whether to send all the discovery information should be sent with the GET request when crawling for auction houses. The argument for sending the information was that then the auction house being queried would not have to query the initiator back, as the information was already sent in the GET. This would reduce the number of requests that had to be sent. The argument against was that the other way would be cleaner and there would be more separation of concerns. The group decided on not sending the information. From our point of view this makes sense, since the system is so small right now, reducing requests is not worth the trade-off.
- Although the teachers recommended we use an existing media-type, the interoperability group decided to implement their own. We decided to do it this way because we couldn't find a suitable predefined media type. From our group's perspective, this is fine since we couldn't find a suitable one, although using a preexisting one would have been better if possible. That would have increased interoperability with external systems.
- Lastly, the interoperability group decided to add a timestamp with auction house information. Since we could be getting information about the same auction house from different nodes in the chain, we need to be able to determine what information is more recent. Having the timestamp allows for this. Our group really likes this decision since it reduces the chance of us using outdated information.

### Task 2: Hypermedia-based Discovery of Auction Houses

#### Implemented discovery functionality:

- Implemented the endpoint that responds to other group's GET requests
- Implemented a web adapter that fetches auction house information from other nodes
- Implemented the crawling algorithm

#### Reasons why discovery mechanism does not impact our architecture (more than just wanting to create a separate service for the crawler):

- The crawler uses synchronous communication to talk with other groups. However, the Crawler will be its own service that asynchronously notifies the Auction House of new discoveries. Therefore, the crawler should not impact the system's overall performance towards its nonfunctional properties

### Task 3: On Decentralization

#### Advantages:

- The fault tolerance is increased as there is no longer a (one or more) centralised points of failure.
- Evolvability is also improved as the tight coupling of clients and servers is relieved
- Scalability is increased as there is less use of intermediary components
- Efficiency is also increased as application-specific interfaces can improve efficiency

**Disadvantages:**

- Adds another dimension that the groups need to make sure works with others
- Crawling a large network can take more resources than just querying one central resource
- Crawling is slightly more complicated to implement than a call to the central resource.  
Especially the focused discovery
- With the current implementation, if the auction house indicated by our seed URI is down or hasn't implemented discovery correctly then we can't find the other auction houses.