

Documentation: Exercise 9

Group 3 – ASSE Autumn Semester 2022

Semantic Hypermedia overlays

We discussed with the other groups what the process of finding the URLs should look like. We have defined that each group gets a "hardcoded" URL of another group as an entry point. This is then the first place we look for the topics we want to subscribe to. If we don't find anything, we go after the URLs of this group and continue searching there until we find what we are looking for. We also discussed the semantics of hypermedia links and defined them as follows:

Link ①	<https://tapas-auction-house.86-119-35-72.asse.scs.unisg.ch/auctions/>; rel="self"
Link ②	<https://switchboard.p3k.io/>; rel="hub"
Link ③	<https://tapas-auction-house.86-119-35-72.asse.scs.unisg.ch/auctions/>; rel="computation"
Link ④	<https://tapas-auction-house.86-119-34-242.asse.scs.unisg.ch/auctions/>; rel="environment"

Link 1 and 2 are the Websub headers according to W3C specification. Link 3 shows the own auction topic in rel + link to our auction feed. Link 4 provides a new resource with the respective topic they will auction.

In our case, we link to our self with the rel="environment" (the topic **we** will launch auctions for) and to group one with the rel="experimental" (the topic **they** will launch auctions for).

Hypermedia-based Discovery of Auction Houses

To discover our "business-clients" we have implemented a basic crawling algorithm.

We have created a new endpoint in our auction house to initiate the crawling. Here an initial resource can be provided which then will be checked first, if no resource is provided, we have configured a default starting point in agreement with the other groups (as mentioned in task 1).

For now, we have also hardcoded the topics we are interested in in our application. When we receive the link information from the initial resource, we check them against

our topic preferences. If they match, we subscribe to the respective feed. Any link that does not match our preferences and is not a WebSub link ("hub" or "self") will then also be crawled.

This process is repeated until we have traversed the entire graph once, or until there are no more links that we have not yet checked.

The crawling algorithm is pretty simple for now and could definitely be improved. However, since we are in a small ecosystem with 4 groups it should be "good enough" for now.

Since the change from a centralized directory to a decentralized crawl for the "business clients" only affected our auction house, we did not see the need to make any significant architectural changes. In fact, only the first step, how we find the feed to subscribe to, changed.

Therefore, we did not write any new ADRs in this exercise.

On Decentralization

At the last exercises and the plug fest we discovered the other "business-partner" through a centralized directory in which every party had to register. Now we have implemented the possibility to search these links with a crawling discovery. The URLs are no longer "hardcoded" in a central registry but are found during the run time. So, we have defined that when searching for the topics entertainment and computation we first query group 1. If they don't have any useful information, we will follow the links we find at group 1. In an iterative process, the corresponding URL should be found. With this decentralization, it would now be fairly easy for new "business partners" to connect to the application and perform tasks for the clients. This allows our decentralized services to coordinate and organize themselves to achieve the action goals. Additionally, we have thus gained an important property, "scalability". This also allows us to decouple from the other group since we no longer need a centralized directory.

Of course, this type of implementation also has disadvantages. For example, the search for the URIs is a lot more complicated to program than a central service. We also can't

understand what exactly is going on with the other "business partners". Maybe they don't execute the task correctly or bid on something they can't execute. Additional functions would also have to be programmed to precisely define the bid process. As the number of "business partners" increases, some of them will be able to execute the same topics. And then a fair process must be found to assign the tasks.

Also, if a "business-client" is not connected in any way to our entry point we will not find it and thus not get the auctions we need.