DISTRIBUTED SYSTEMS

Trabalho 1

Nuno Preguiça, Sérgio Duarte, João Resende

PROJECT

Backend of Mastodon-like system.

https://mastodon.social

Mastodon is like a peer-to-peer Twitter.

A user is registered in a server/domain. A user can follow users from its server/domain or other servers/domains.

A user has a feed of messages,

A message posted by user is added to its own feed and to the feed of other users that follow her.

PROJECT - ARCHITECTURE

System composed by multiple domains.

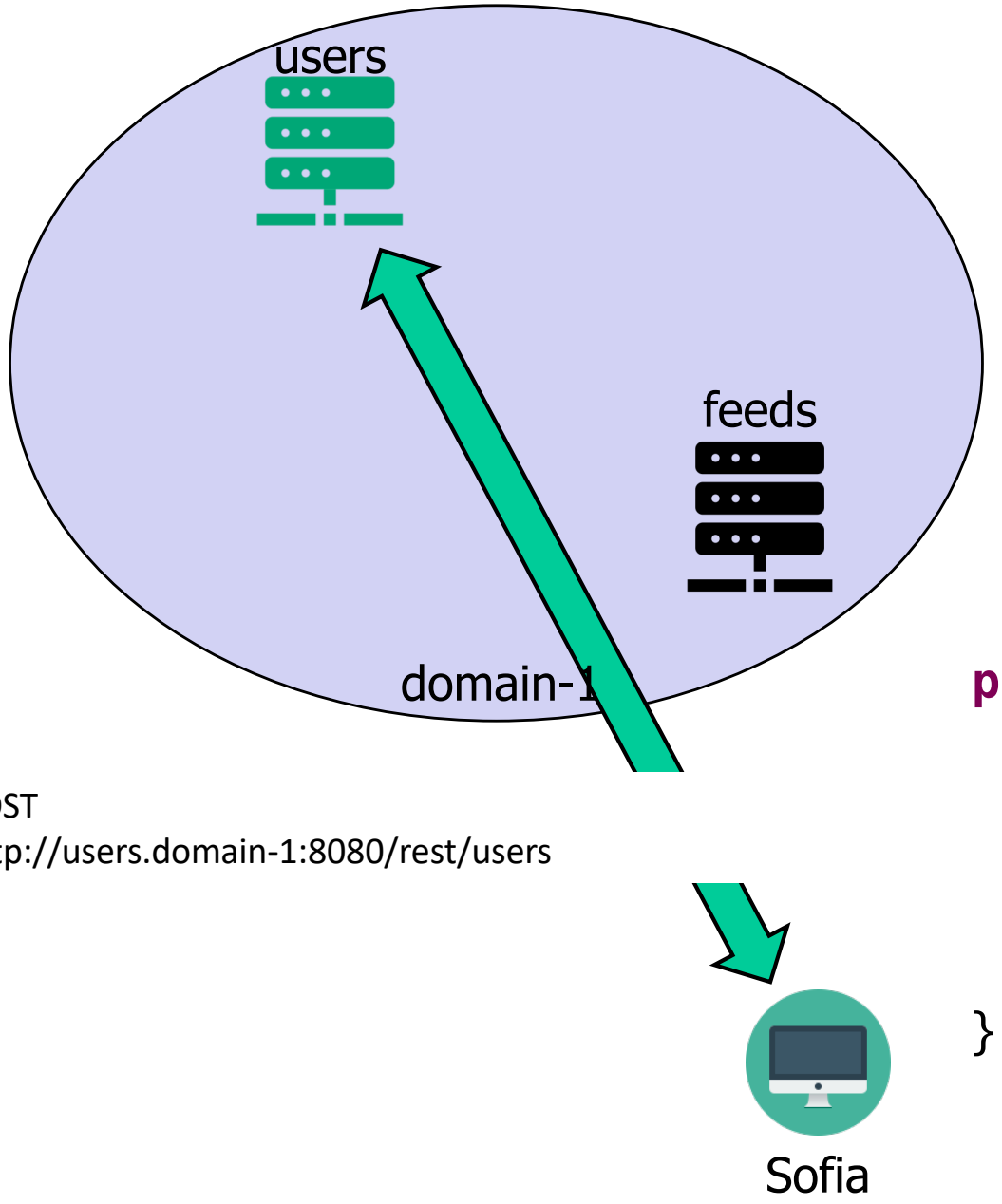
Each domain includes two services: users, messages.

Service **users** maintains information about users in that domain.

Service **feeds** maintains the feeds of users in that domain.

EXECUTING THE CREATE USER

Client calls REST endpoint

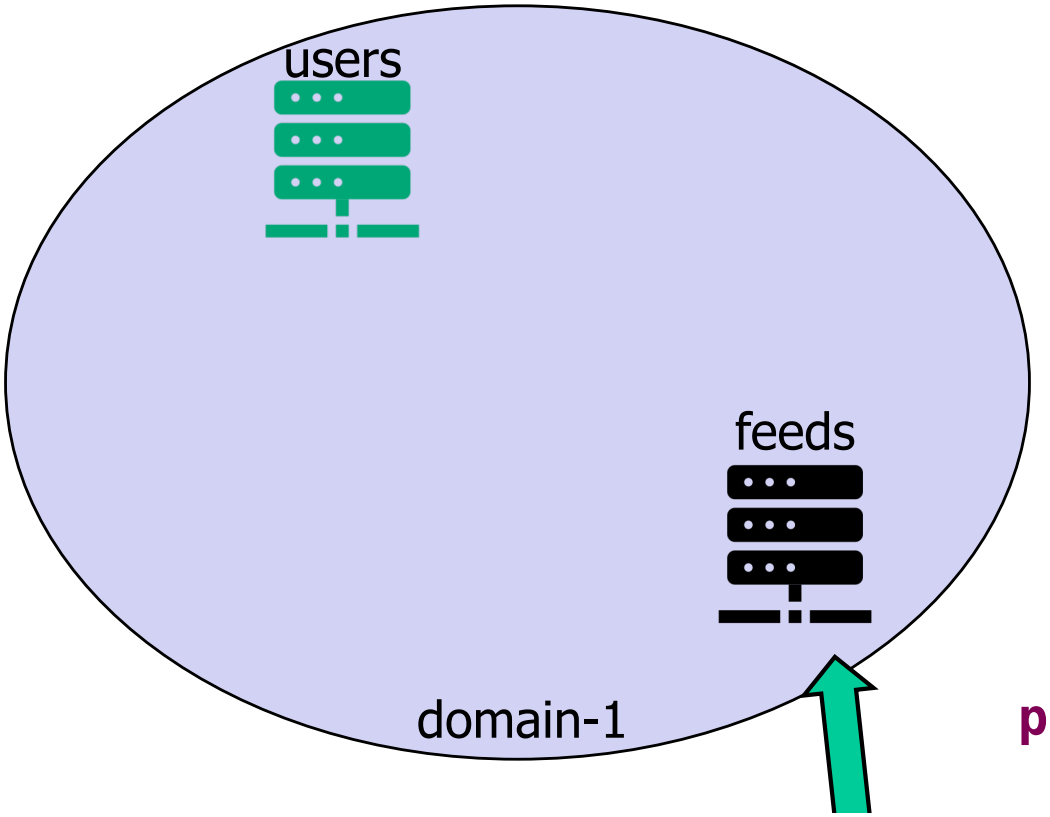


POST
http://users.domain-1:8080/rest/users

```
public class User {
    private String name;
    private String pwd;
    private String displayName;
    private String domain;
    …
}
```

EXECUTING POST MESSAGE

Client calls REST endpoint



POST
http://feeds.domain-1:8080/rest/feeds/sofia@domain-1?pwd=Rf&4

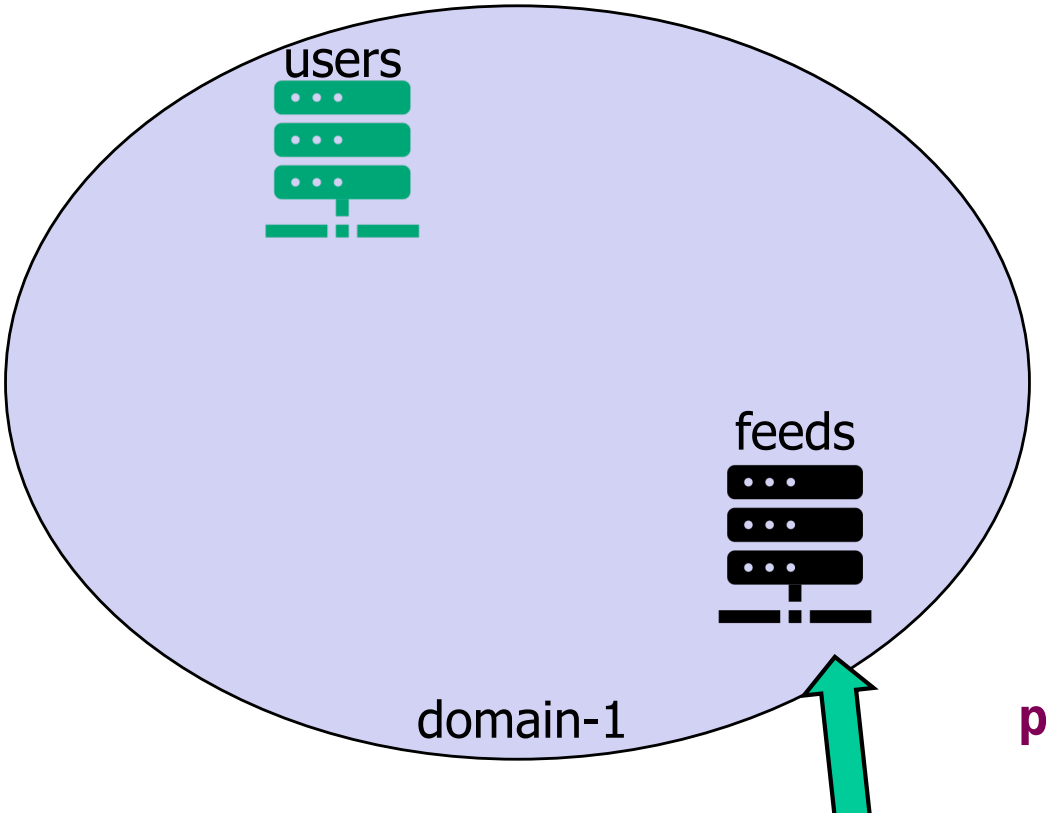
Sofia

```
public class Message {
    private long id;
    private String user;
    private String domain;
    private long creationTime;
    private String text;
    ...
}
```

EXECUTING POST MESSAGE

How is the request processed?

1. Check that inputs are correct (incl. checking password of the owner and domain in the message)

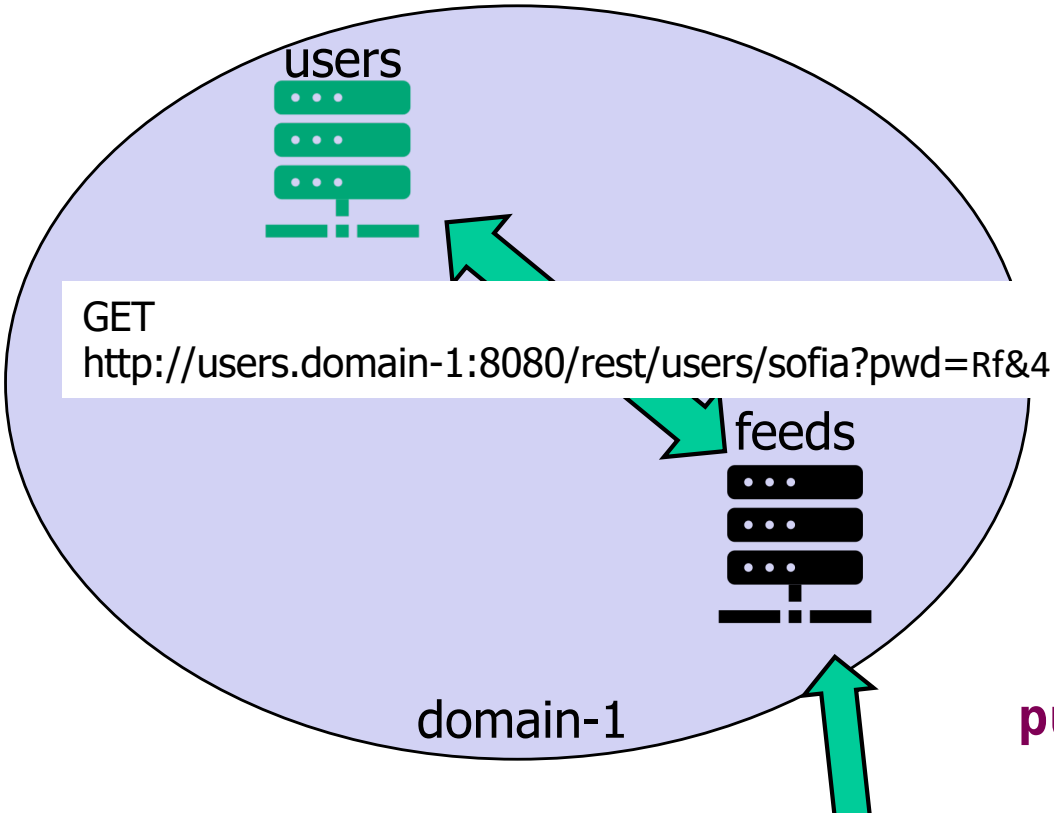


POST
http://feeds.domain-1:8080/rest/feeds/sofia@domain-1?pwd=Rf&4

Sofia

```
public class Message {
    private long id;
    private String user;
    private String domain;
    private long creationTime;
    private String text;
    ...
}
```

EXECUTING POST MESSAGE



POST
http://feeds.domain-1:8080/rest/feeds/sofia @domain-1?pwd=Rf&4

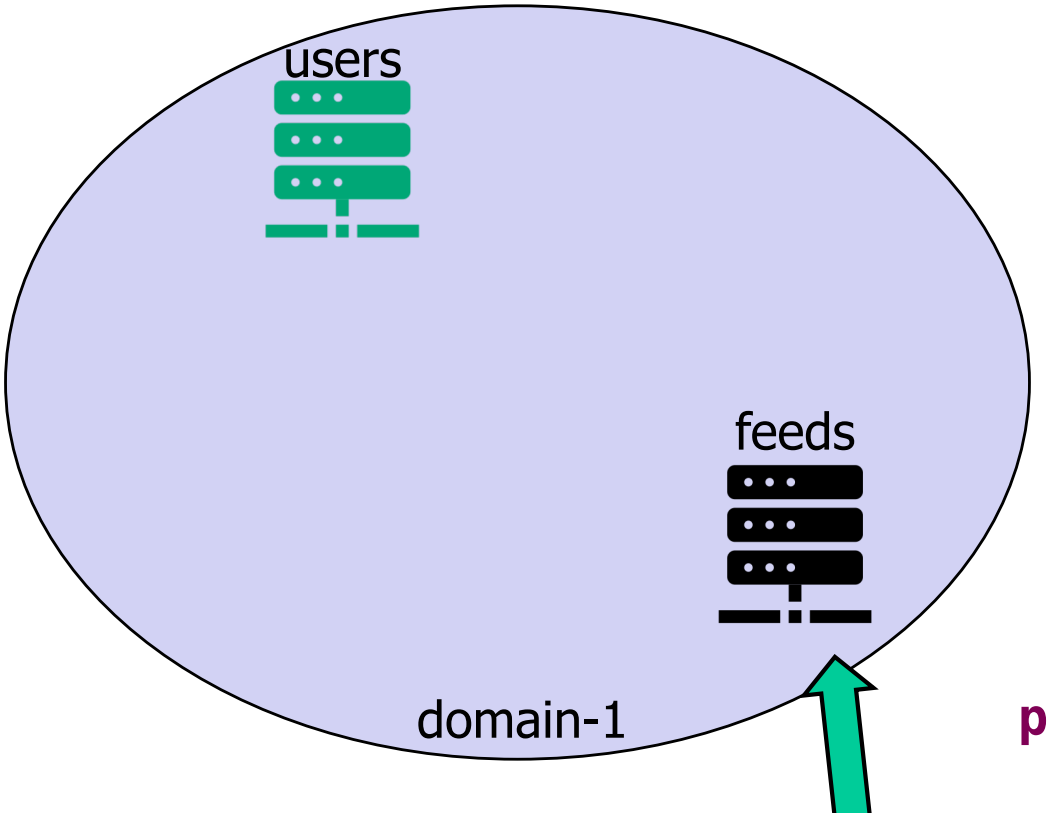
Sofia

How to check the password?

Need to call users service to check if password is correct. E.g. can use the getUser endpoint or a new one

```
public class Message {
    private long id;
    private String user;
    private String domain;
    private long creationTime;
    private String text;
    ...
}
```

EXECUTING POST MESSAGE



POST
http://feeds.domain-1:8080/rest/feeds/sofia @domain-1?pwd=Rf&4

Sofia

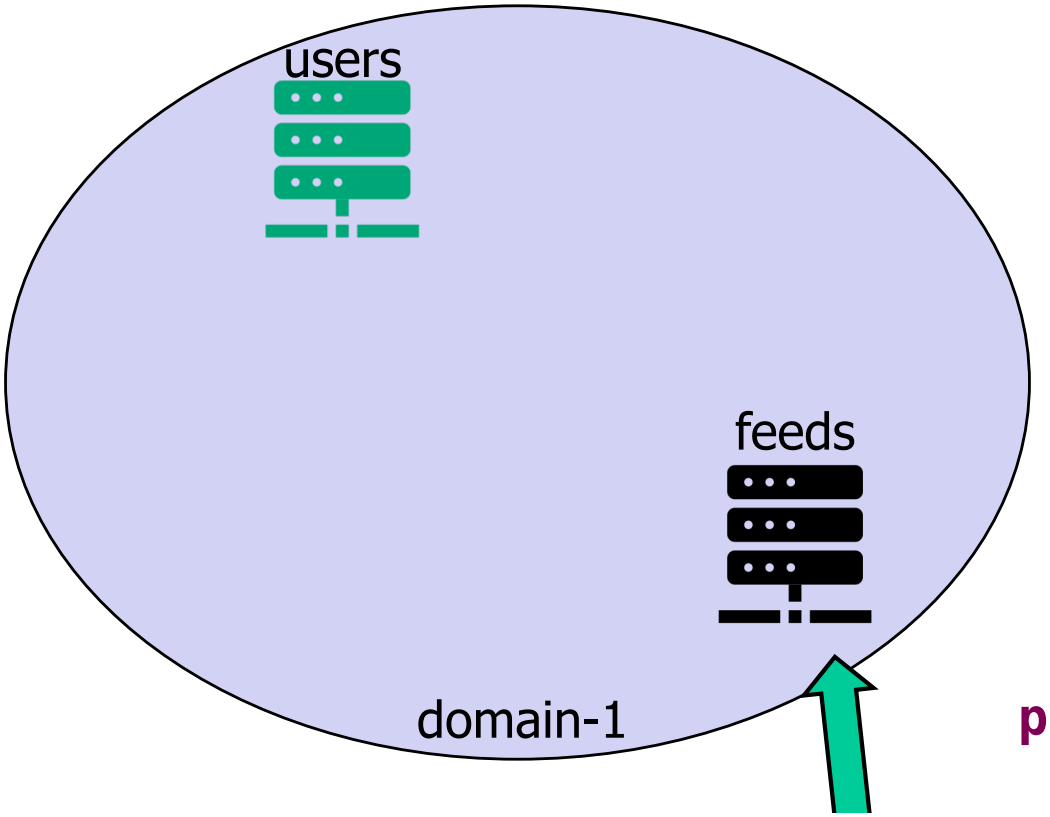
How is the request processed?

1. Check that inputs are correct (incl. checking password of the owner and domain in the message)

2. If **not OK**, return appropriate HTTP error message.

```
public class Message {
    private long id;
    private String user;
    private String domain;
    private long creationTime;
    private String text;
    ...
}
```

EXECUTING POST MESSAGE



POST
http://feeds.domain-1:8080/rest/feeds/sofia@domain-1?pwd=Rf&4

Sofia

How is the request processed?

1. Check that inputs are correct (incl. checking password of the owner and domain in the message)

2. If **OK**, assign id and creationTime to the message and store it to the users feeds.

```
public class Message {
    private long id;
    private String user;
    private String domain;
    private long creationTime;
    private String text;
    ...
}
```

How are messages stored?

Alternative 1

- When there is a post message, the message is stored in all feeds that follow the user.
- The get messages operation simply returns the messages stored in the feed.

Alternative 2

- A message is stored only in the feed of the user that posts the message.
- The get messages of a user needs to retrieve messages from all feeds that the user follows.

How are messages stored? Challenges.

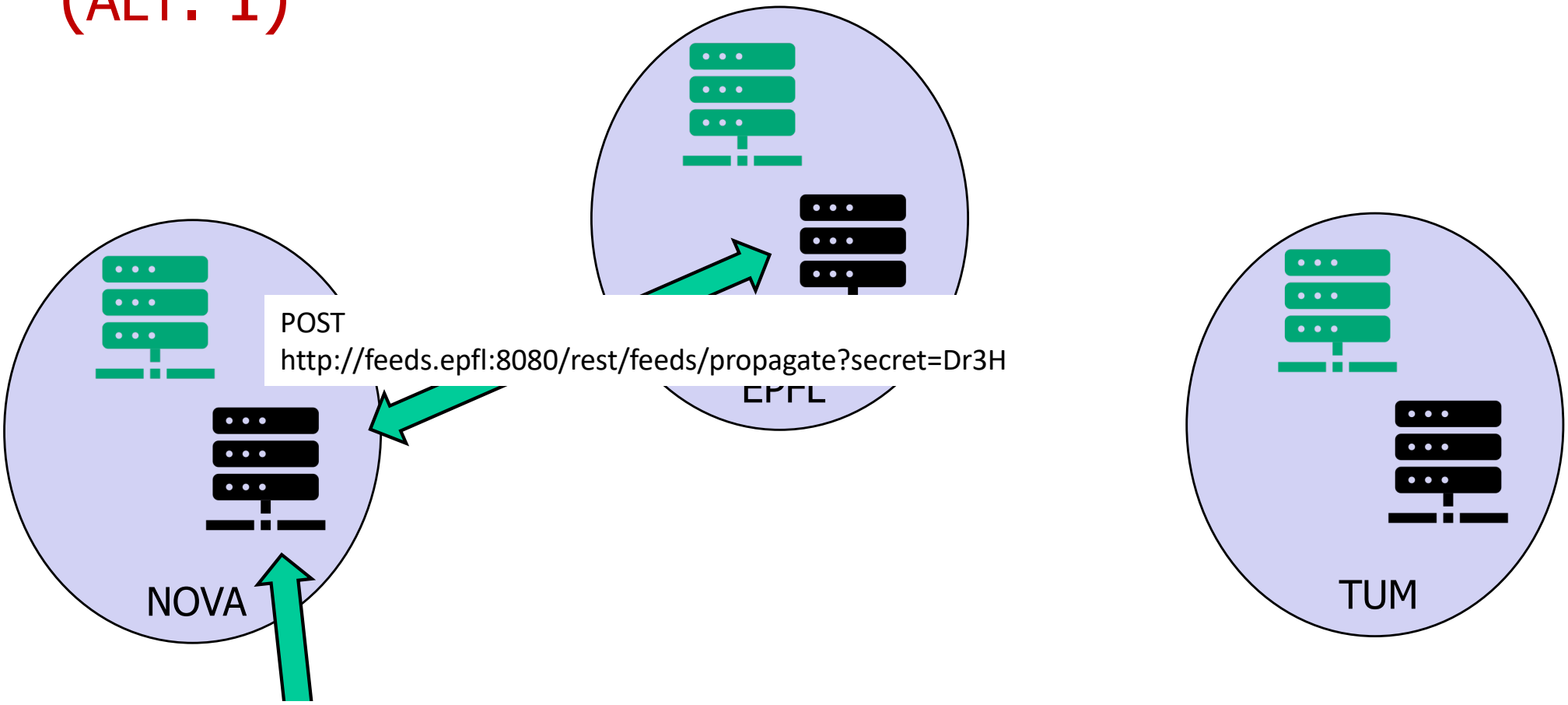
Alternative 1

- In the post message, it is necessary to handle communication faults. E.g. background thread to propagate messages that failed.

Alternative 2

- In the get messages, it is necessary to think about efficiency (e.g. by caching messages) – otherwise, operation is just too slow.

Executing post Message with multiple domains (alt. 1)



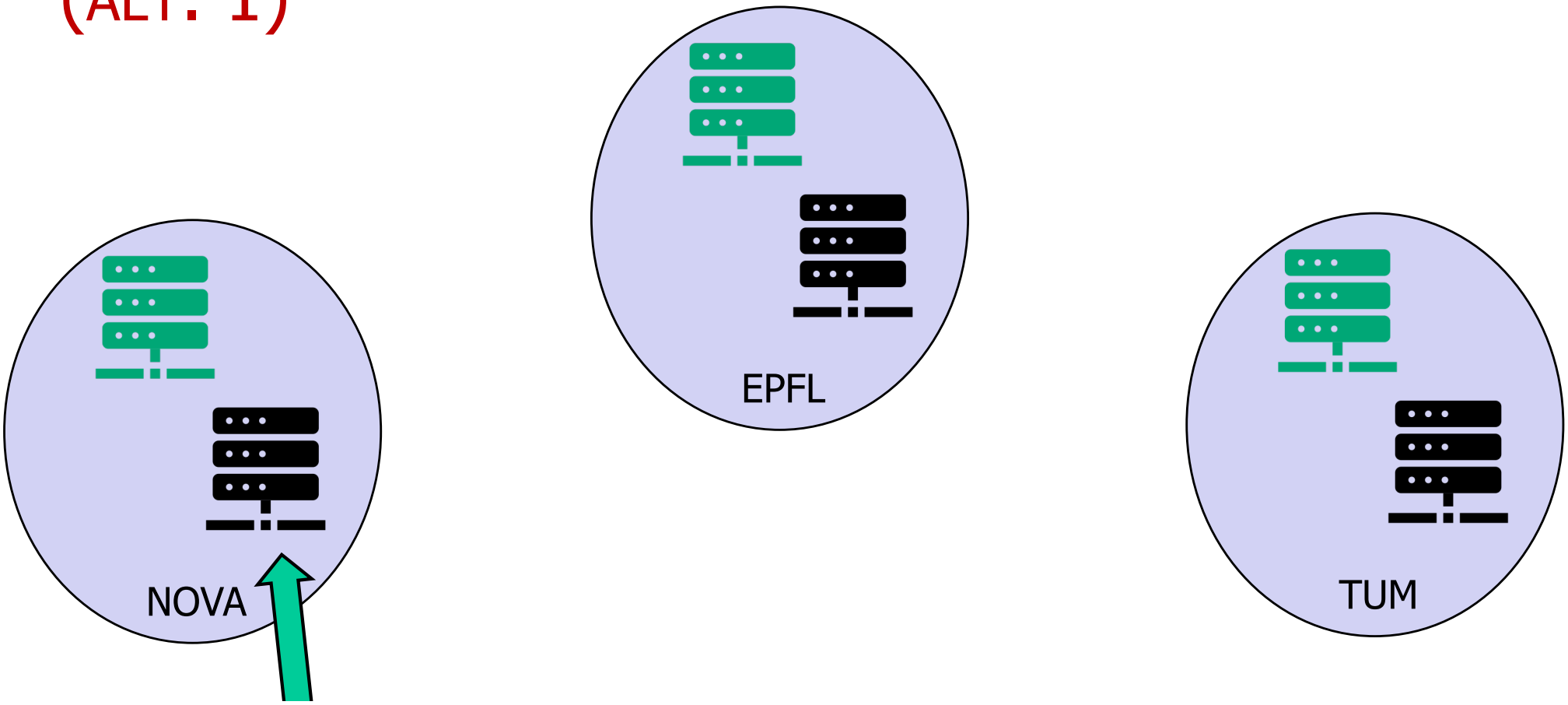
POST
http://feeds.nova:8080/rest/feeds/sofia@nova?pwd=Rf&4

Sofia

Assuming there is at least one user at EPFL that follows sofia.

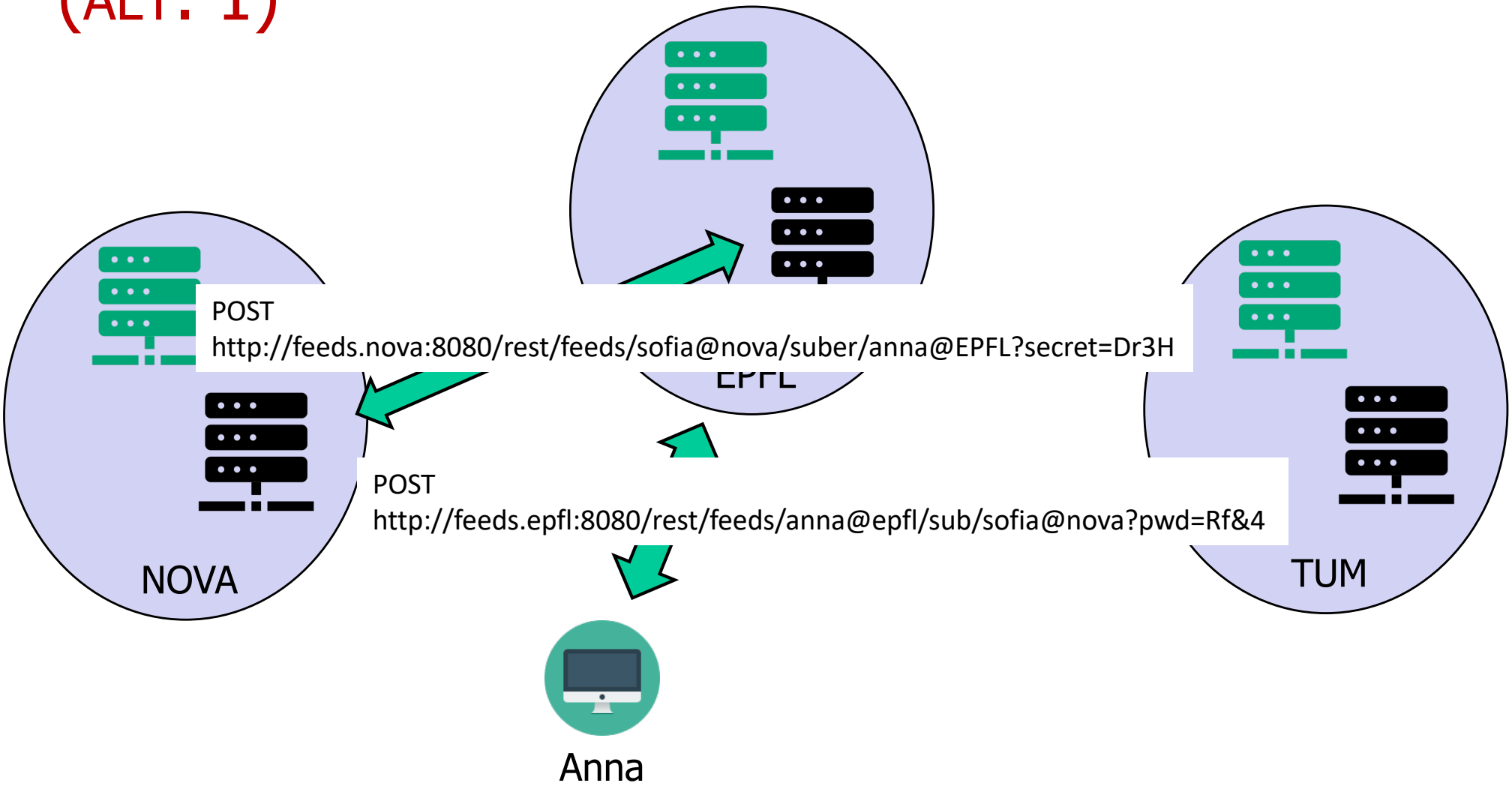
Need to create a special method to propagate message… the message only needs to be sent once even if there are more that one user that follows sofia – secret is a secret known by servers to authenticate each other.

Executing get Message with multiple domains (alt. 1)



Get messages operation is local.

EXECUTING FOLLOW USER WITH MULTIPLE DOMAINS (ALT. 1)



Needs to propagate information to NOVA that there is someone (anna) at EPFL that follows sofia.

INTERFACES AND REMAINING OPERATIONS

Check the course's web page for more information.

Important note: your servers must implement the interface as defined, and return the appropriate error messages – you cannot change the provided methods, but you can add new ones.