Interactions Meeting Notes

March 2025 (onsite at NetApp)

**Interactions – what is it?**

Goal here is to create a standard for communication between the various areas:

- Backend
- Front end
- Security
- API

Interactions will describe what NEEDS to be done, but not HOW to do it.  For example:

- Mapping
- Protocol translation
- Permissions

API group responsible for external interactions, such as:

- Security
- Content management
- Each area will have overlap with the other groups
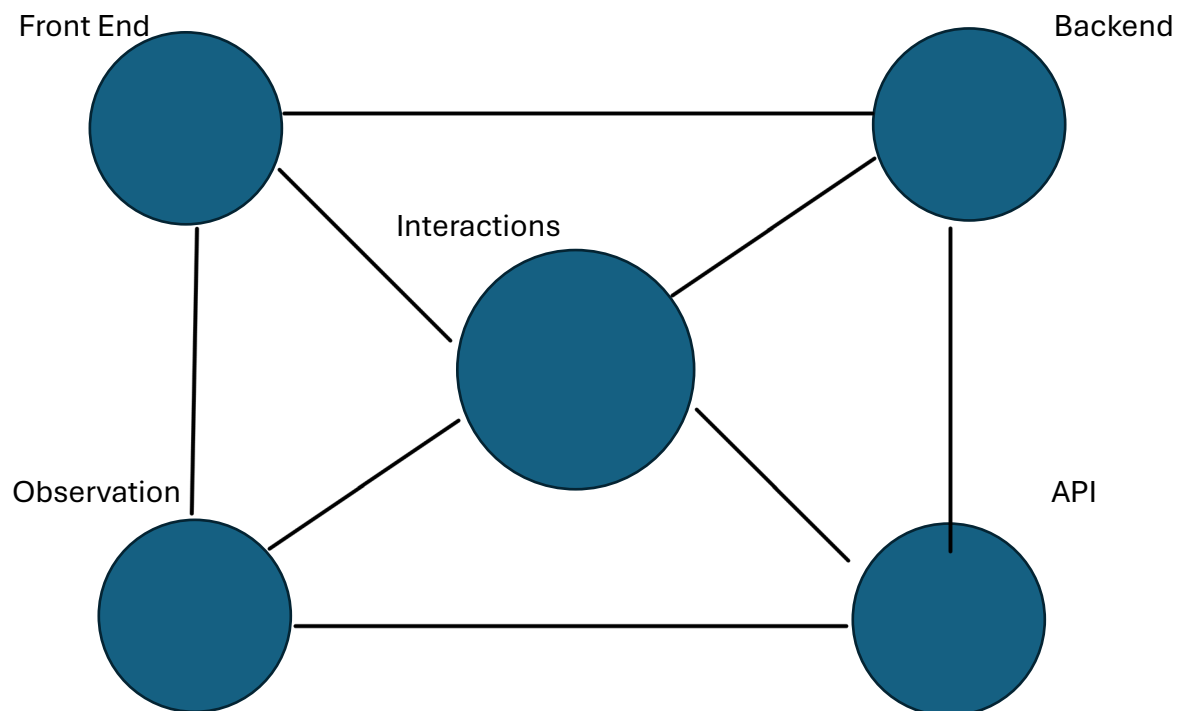
Ideas on how to structure this:

- Define what each of these are
- Define what the interactions are between these

How?

Define using a workflow-based approach, such as:

Ask the backend (BE): who is this guy? What's he requesting? Does he have permission?

What are some specific examples?

- Permissions
- Regulatory (industry specific, political/geographic, etc.)
- Is it synchronous or asynchronous?
- How/what is fetched/pre-fetched?

Define Types of interactions

Define types of workflows

Define types of relationships

- Local dataset
- Remote dataset

Hierarchy?

- First ask about permissions/security
- Then ask the backend – what are your capabilities?
- Then ask the front end – what are your requests and/or capabilities?

(continues)

**Control Plane** (manifestation of Interactions) – use a control plane as a means of managing interactions

To illustrate, the control plane (management function) is within the Interactions sphere
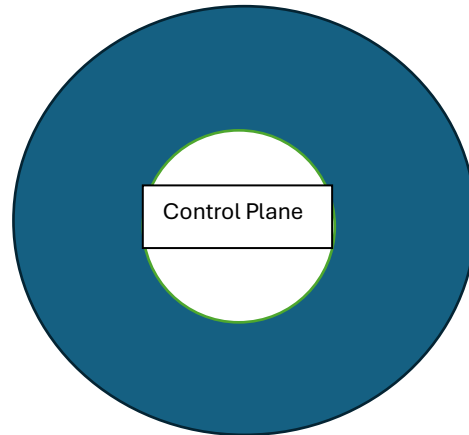


The observation group is working on security (monitoring), while the interactions group is working on management.

**Workflows**

Creating relationships using a workflows-based approach seems to be the best way to conceptualize how to proceed.

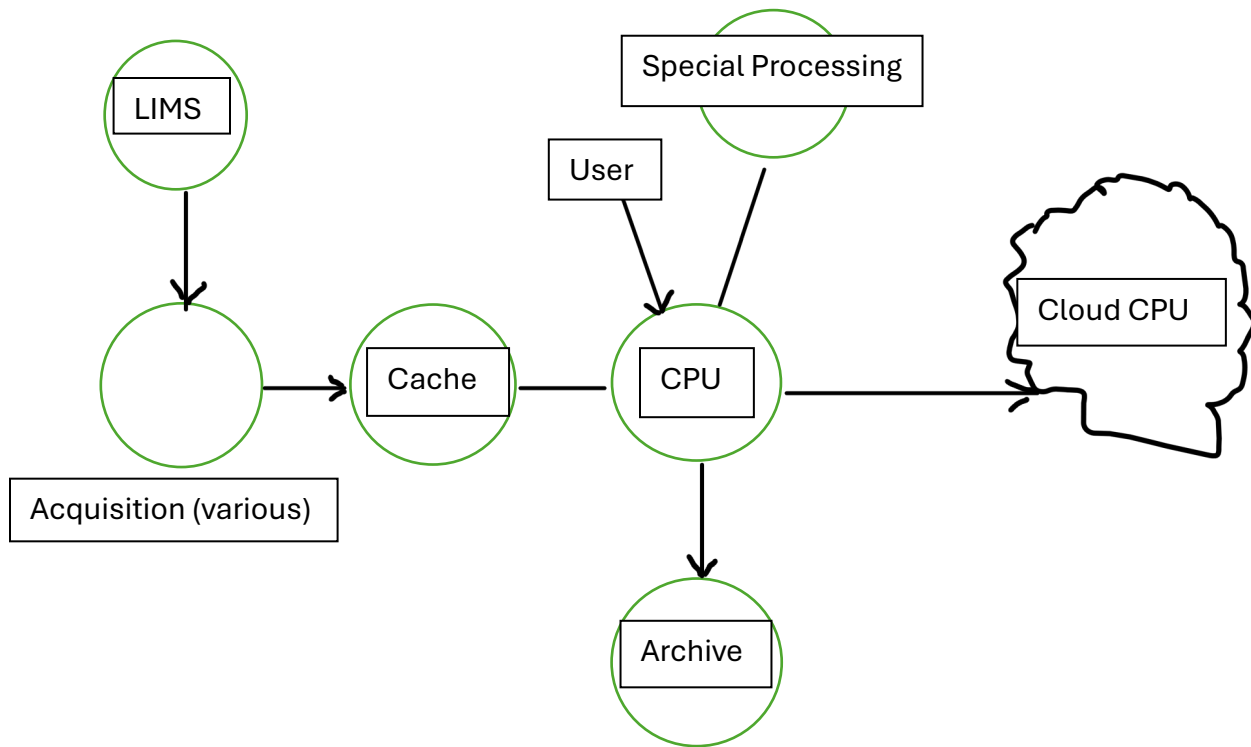Create relationships between the control plane (CP) and:

- Front end
- Back end
- Type (file, object, etc)

If starting with a user, the process will look like:

- User to control plane
- Control plane to security
- Control plane to back end
- Control plane to front end

**Constantine's Use Case**

Looking at process from initiation to target – data flow use case:



Actions:

- Create
- Destroy
- Modify

Workflows

      Relationships

           Interactions