2 Modeling Agents and Organizations

Objective:  
To model the relationship between a specific person, their role, and their organization, while also correctly modeling their names and identifiers according to rigorous ontological principles.

Scenario:  
An employee, whose name is "Jane Doe," holds the role of "Senior Systems Engineer." She performs this role for the "F-35 Lightning II Program."

Description:  
This pattern now combines the agent-role model with the information-entity model.

1. **The Agents and Role:** The core structure remains the same: an iof:Person (ns:employee-789) iof:hasRole an iof:OccupationRole (ns:engineer-role-456), which iof:hasOrganizationalContext an iof:Organization (ns:f35-program).
2. **Modeling the Names (The Rigorous Way):** We do not attach the literal string "Jane Doe" directly to the person. Instead:
   * The person ns:employee-789 is the subject of some information.
   * The abstract information is the name itself (ns:jdoe-person-name). This is an iof:DesignativeInformationContentEntity. It iof:is\_about the person ns:employee-789.
   * This abstract name is written down or stored somewhere. That "somewhere" is the iof:InformationBearingEntity (e.g., a specific field in an HR database, ns:hr-db-field-name-1138).
   * Finally, this information bearer is linked to the actual text string "Jane Doe" using the iof:has\_text\_value data property.

The same detailed pattern applies to the name of the organization and the description of the role. This method allows us to track *where* we got the name from (e.g., the HR database vs. a project charter document), which is impossible with a simple rdfs:label.

A diagram of a company

AI-generated content may be incorrect.

Diagram: Employee Role with Rigorous Naming