University of Notre Dame

CENTER FOR RESEARCH COMPUTING

An Ontology Design Pattern towards Preservation of Computational Experiments

Da Huo, Jaroslow Nabrzyski, Charles F. Vardeman II

University of Notre Dame

Department of Computer Science and Engineering

Center of Research Computing

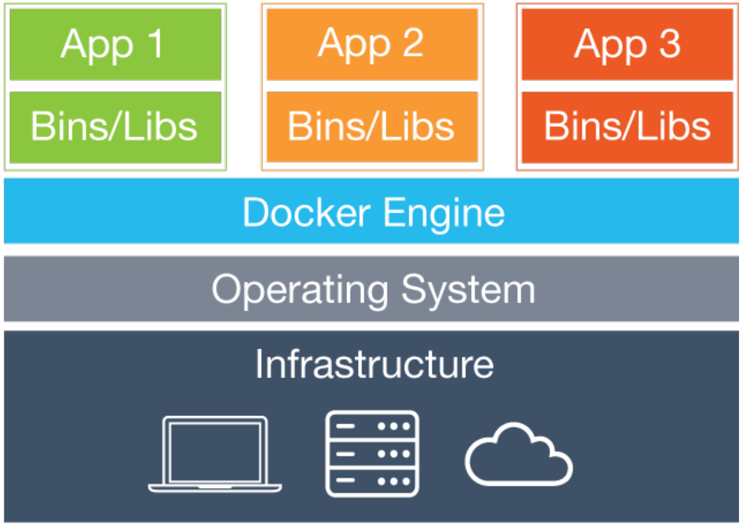
dhuo@nd.edu

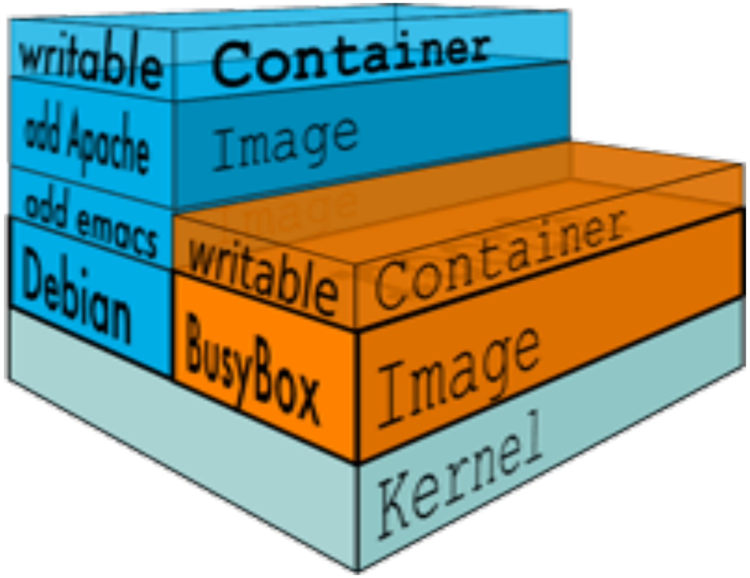
Introduction

- Demand for an efficient approach to preserve and share experiments.
- Preserve Computational Experiment - Replication of numeric results vs. Context of the calculation.
- Preservation should be described both in machine and human readable fashions.
- Smart Container(SC) ontology towards conceptualizing computational experiments from the perspective of computational environments and activities within using Docker LXC as a preservation tool.

docker Background

- Docker Linux container as a scaffold
  - Light-weighted virtualization platform
  - Versioned file system
  - Modular design for distribution of software component
  - A sustainable community(industry, CERN[2])

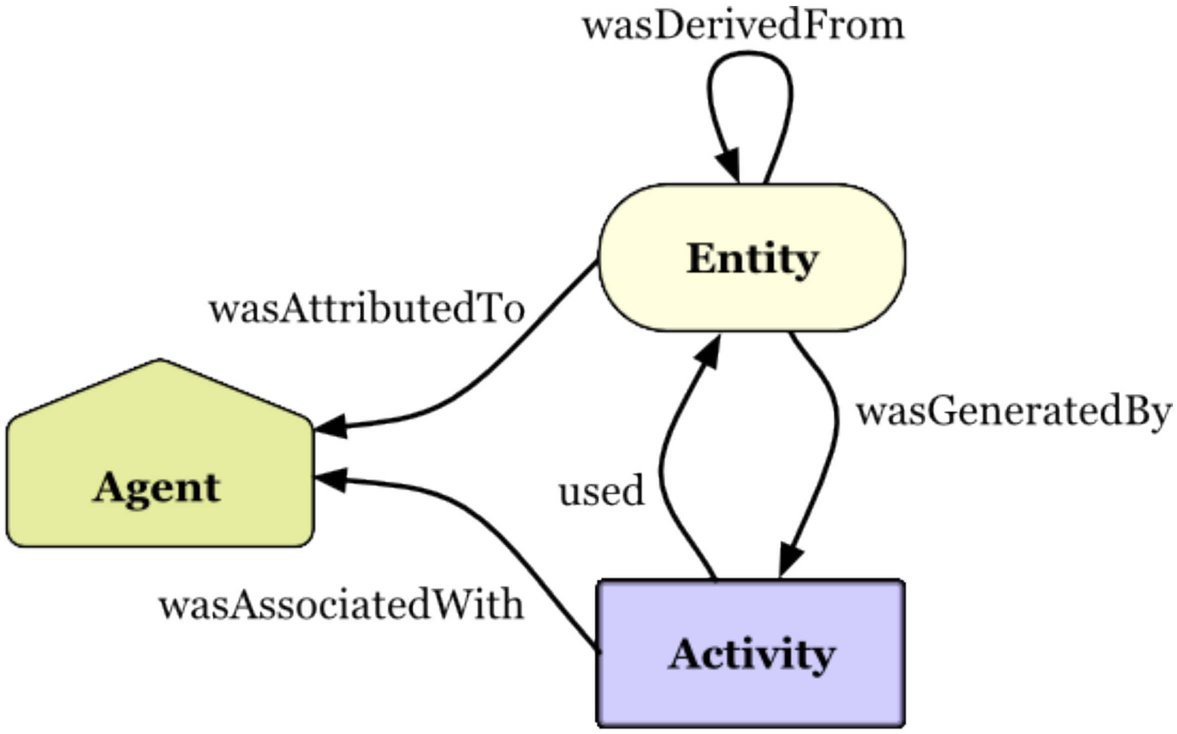




Background

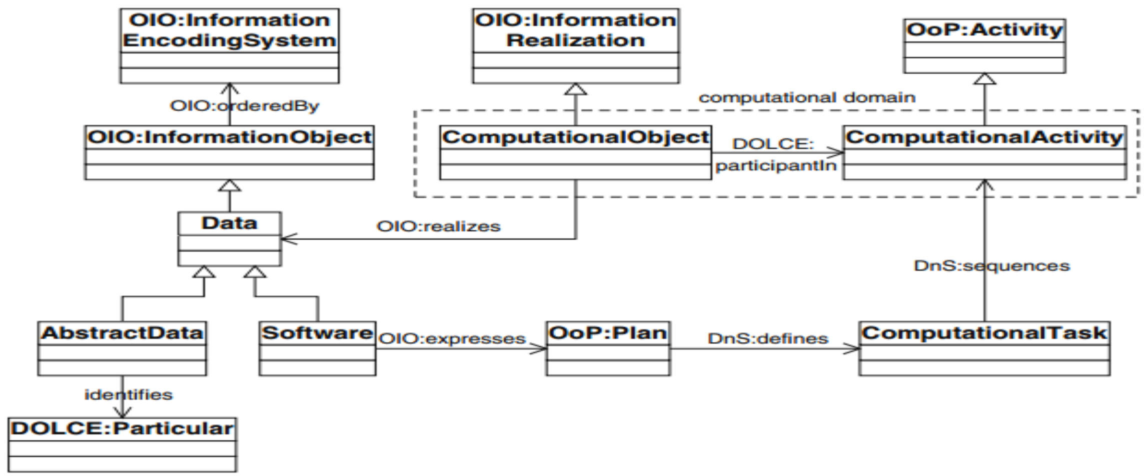
We referenced or aligned existing ontologies and patterns, such as PROV-O, CSO and ACT for discoverability, interoperability, query-ability and future extensibility.

PROV-O

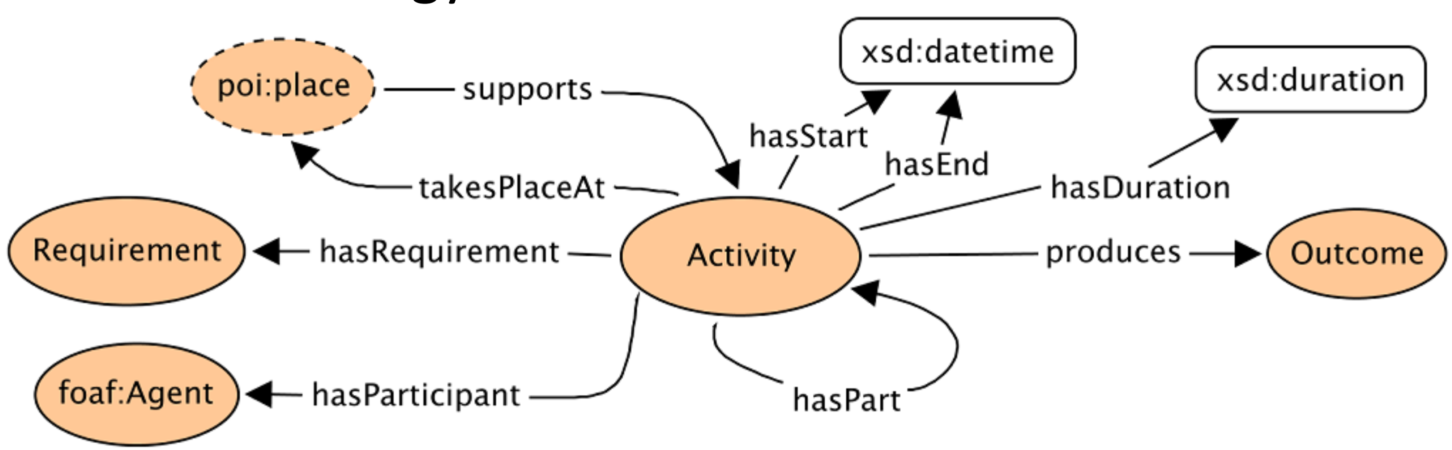


Background

Core Software Ontology



Activity Pattern Ontology

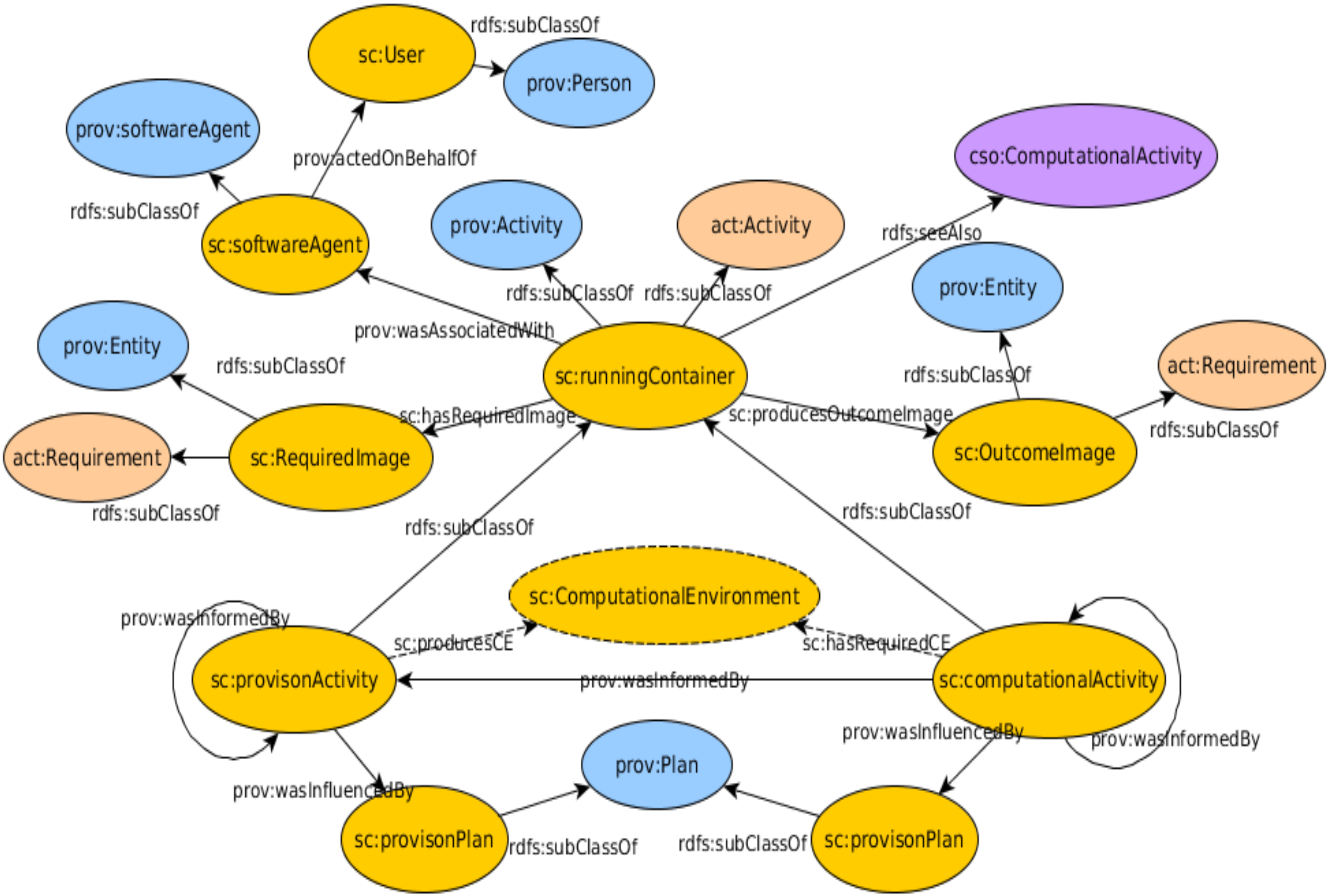


Toward the Formalization of "Smart Containers"

Some competency questions:

1) "What are the requirements for a computational activity?"
2)"What was the environment in which the activity was performed in terms of software components?"
3)"What is the order in which provisioning activities must occur?"
4) "What software agents are responsible for a particular result or outcome".

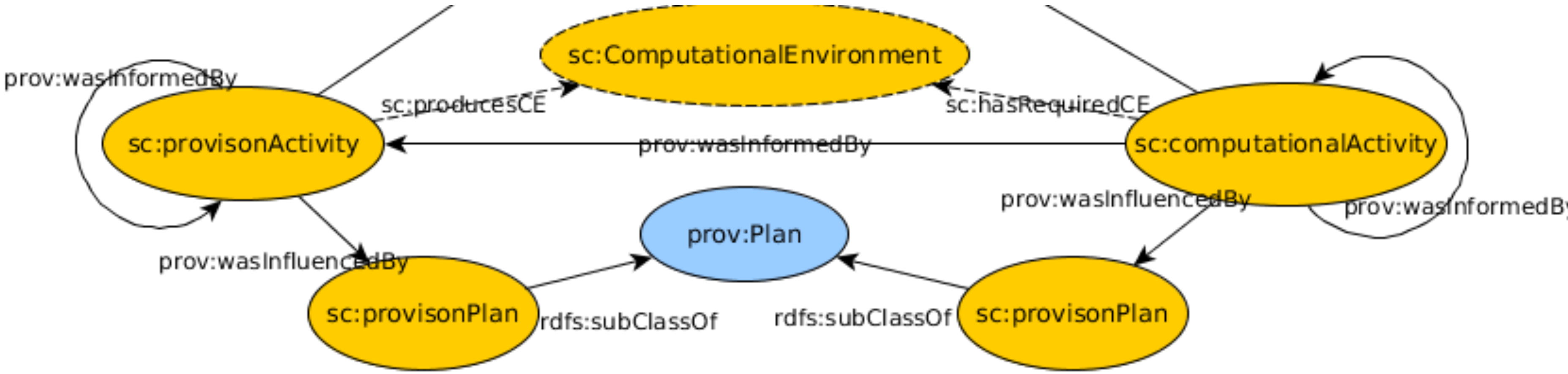
- Using a modular approach by **systematic** alignment of concepts present in Docker as a computational environment.
- Reusing vocabulary terms where possible to contextualize computational activities.
- Facilitating control over unintended consequences of large ontology and entailment inconsistences


sc:User
rdfs:subClassOf
prov:Person
prov:softwareAgent
prov:actedOnBehalfOf
cso:ComputationalActivity
rdfs:subClassOf
sc:softwareAgent
prov:Activity
act:Activity
rdfs:seeAlso
prov:Entity
rdfs:subClassOf
rdfs:subClassOf
prov:wasAssociatedWith
prov:Entity
rdfs:subClassOf
sc:runningContainer
rdfs:subClassOf
act:Requirement
sc:hasRequiredImage
sc:producesOutcomeImage
rdfs:subClassOf
act:Requirement
sc:RequiredImage
sc:OutcomeImage
rdfs:subClassOf
rdfs:subClassOf
rdfs:subClassOf
sc:ComputationalEnvironment
prov:wasInformedBy
sc:producesCE
sc:hasRequiredCE
sc:provisonActivity
prov:wasInformedBy
sc:computationalActivity
prov:wasInfluencedBy
prov:wasInformedBy
prov:Plan
prov:wasInfluencedBy
sc:provisonPlan
rdfs:subClassOf
rdfs:subClassOf
sc:provisonPlan

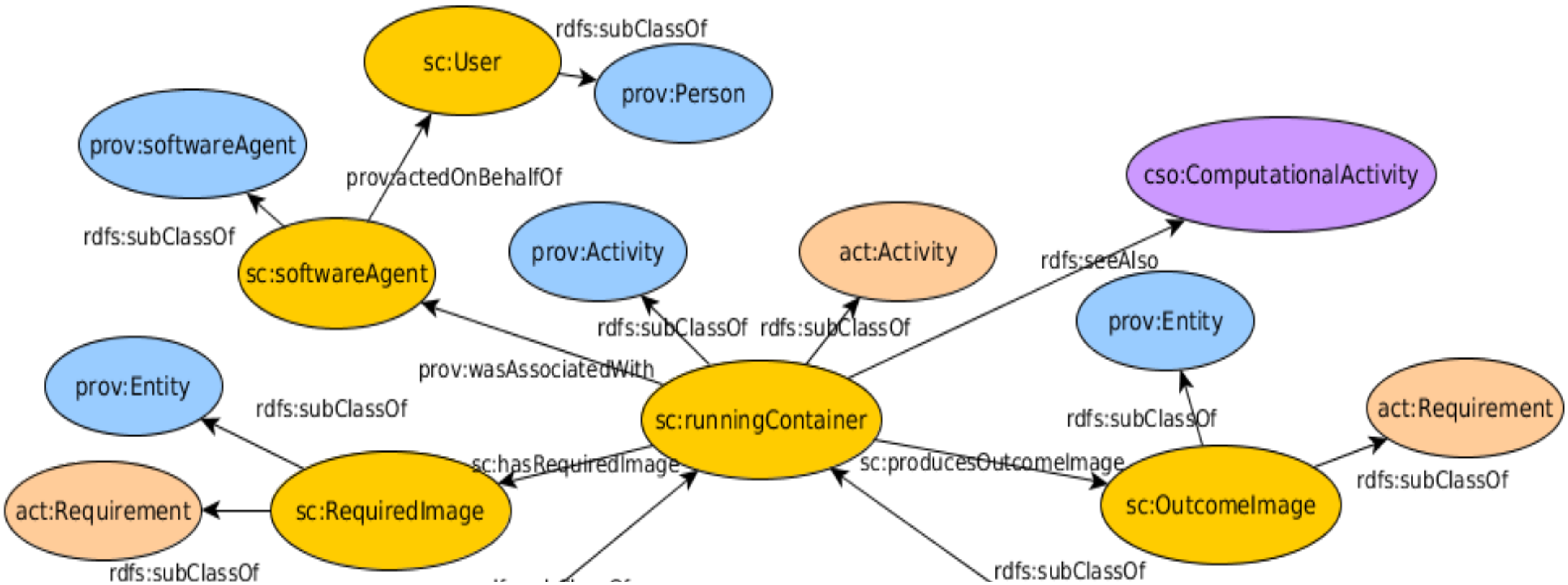
provisioning activities: create an appropriate environment for computational activities. A sequences of provisioning activities was planned by a **provision plan.**

computational activities: directly produce scientific observations and affected by a **workflow plan**

We propose align both with **prov:Activity**



runningcontainer: a Docker LXC concept that represents an activity **hasRequiredImage** and **producesImage**, also **was Associated With** a **software Agent**, which **actedOnBehalfOf** a **User**
use **rdf:seeAlso** to reference the **ComputationalActivity** concept from Core Software Ontology to **runningContainer**



Conclusion

An ontology design pattern contextualizing computational experiments using Docker as an infrastructure example.

- **Capture existing scientific workflow frameworks and descriptions with a Docker LXC**
- **Data to be integrated in a consistent manner**
- **A common description of computational environment**

Analyzed main concepts in computational experiment and aligned with PROV-O, CSO and ACT for wider extensions.

Ultimate goal:
An automated tool and infrastructure.

University of Notre Dame

CENTER FOR RESEARCH COMPUTING

Reference

- 1. Abdalla, A., Hu, Y., Carral, D., Li, N., Janowicz, K.: An ontology design pattern for activity reasoning
- 2. Compton, M., Corsar, D., Taylor, K.: Sensor data provenance: Ssno and prov-o to- gether at last. In: To appear 7th International Semantic Sensor Networks Workshop (October 2014) (2014)
- 3. Gangemi, A.: Ontology design patterns for semantic web content. pp. 262–276. Springer (2005), http:// link.springer.com/chapter/10.1007/11574620_21
- 4. Janowicz, K., Hitzler, P., Adams, B., Kolas, D., Vardeman II, C.: Five stars of Linked Data vocabulary use. Semantic Web http://iospress.metapress.com/ index/053766UR810L7274.pdf
- 5. Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., Gar- ijo, D., Soiland-Reyes, S., Zednik, S., Zhao, J.: Prov-o: The prov ontology. W3C Recommendation 30 (2013)
- 6. Ma, X., Zheng, J.G., Goldstein, J.C., Zednik, S., Fu, L., Duggan, B., Aulenbach, S.M., West, P., Tilmes, C., Fox, P.: Ontology engineering in provenance enablement for the national climate assessment 61, 191–205, http:// linkinghub.elsevier. com/retrieve/pii/S1364815214002254
- 7. Oberle, D., Grimm, S., Staab, S.: An ontology for software. In: Handbook on on- tologies, pp. 383–402. Springer (2009)

Acknowledgement We acknowledge funding from NSF grant PHY-1247316 “DAS- POS: Data and Software Preservation for Open Science” and the Center for Research Computing at UND. We also acknowledge helpful conversations with Michelle Cheatham, Stian Soiland-Reyes, Matthew Gamble and Carole Goble.

DASPOS Data and Software Preservation for Open Science