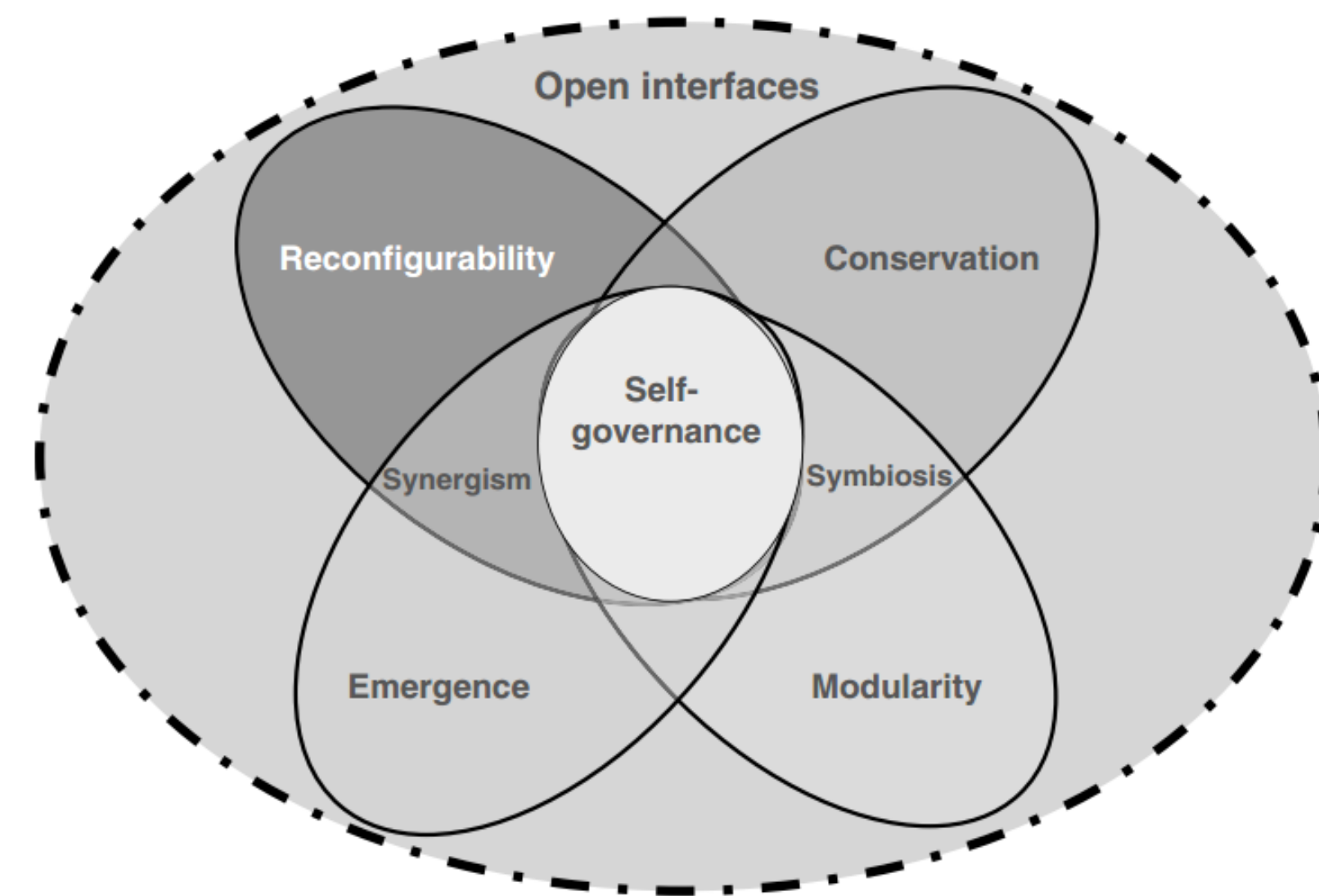


## System of Systems Engineering Principles and Methodology

### Open System Principles

SoS Engineering can capitalize on the open systems strategy to build evolvable, affordable, and sustainable systems:

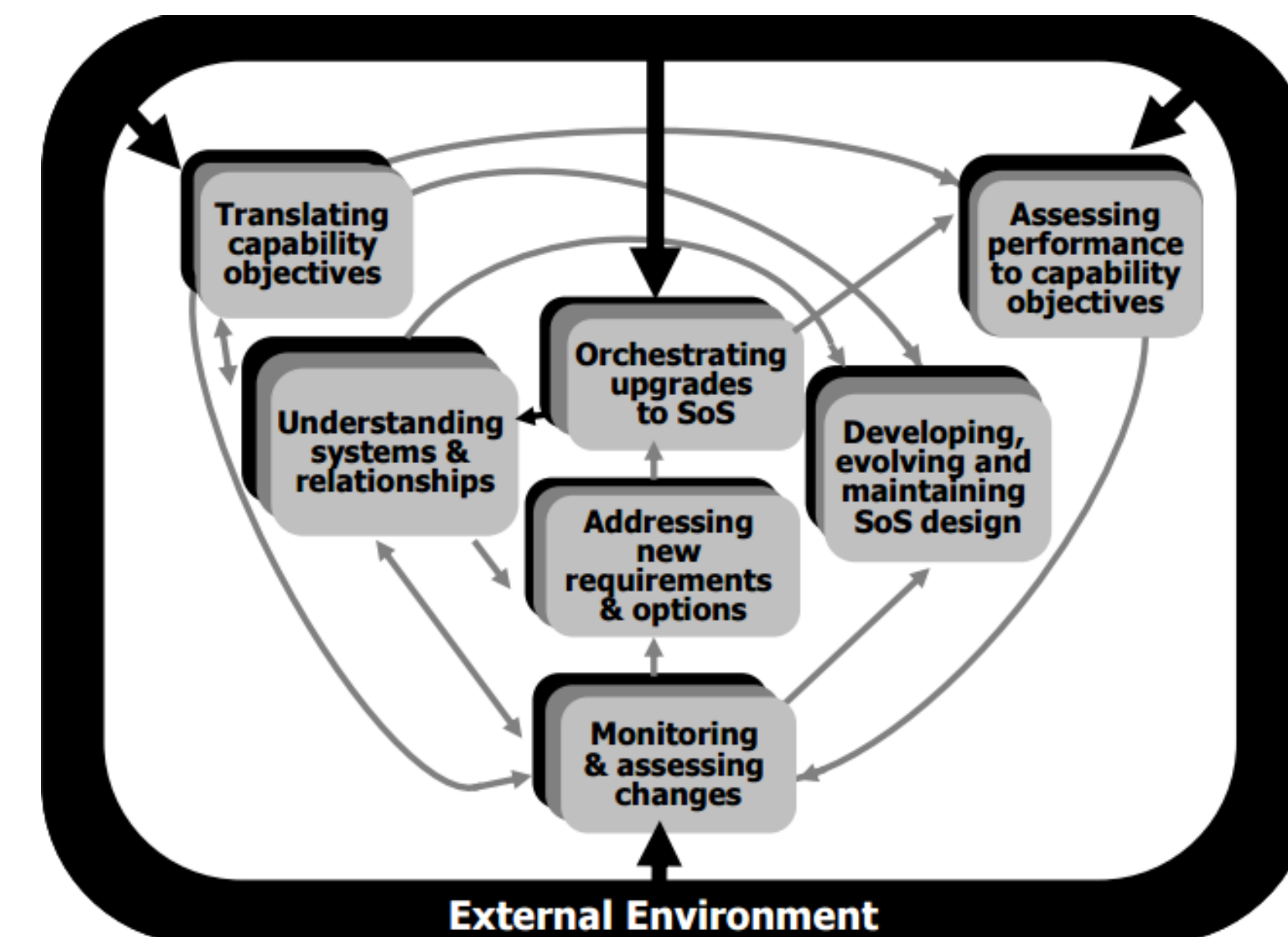


Anani, C. (2009). An Open Systems Approach to System of Systems Engineering. In M.Jamshidi (Ed.), *System of systems engineering - Innovations for the 21st century*. (pp. 21-43). Hoboken, N.J.: Wiley

**System of Systems (SoS)** is a complex system consisting of components which are *independent* systems themselves, achieving a *unique end-state* by *providing synergistic capability* from its component systems. Its characteristics include:

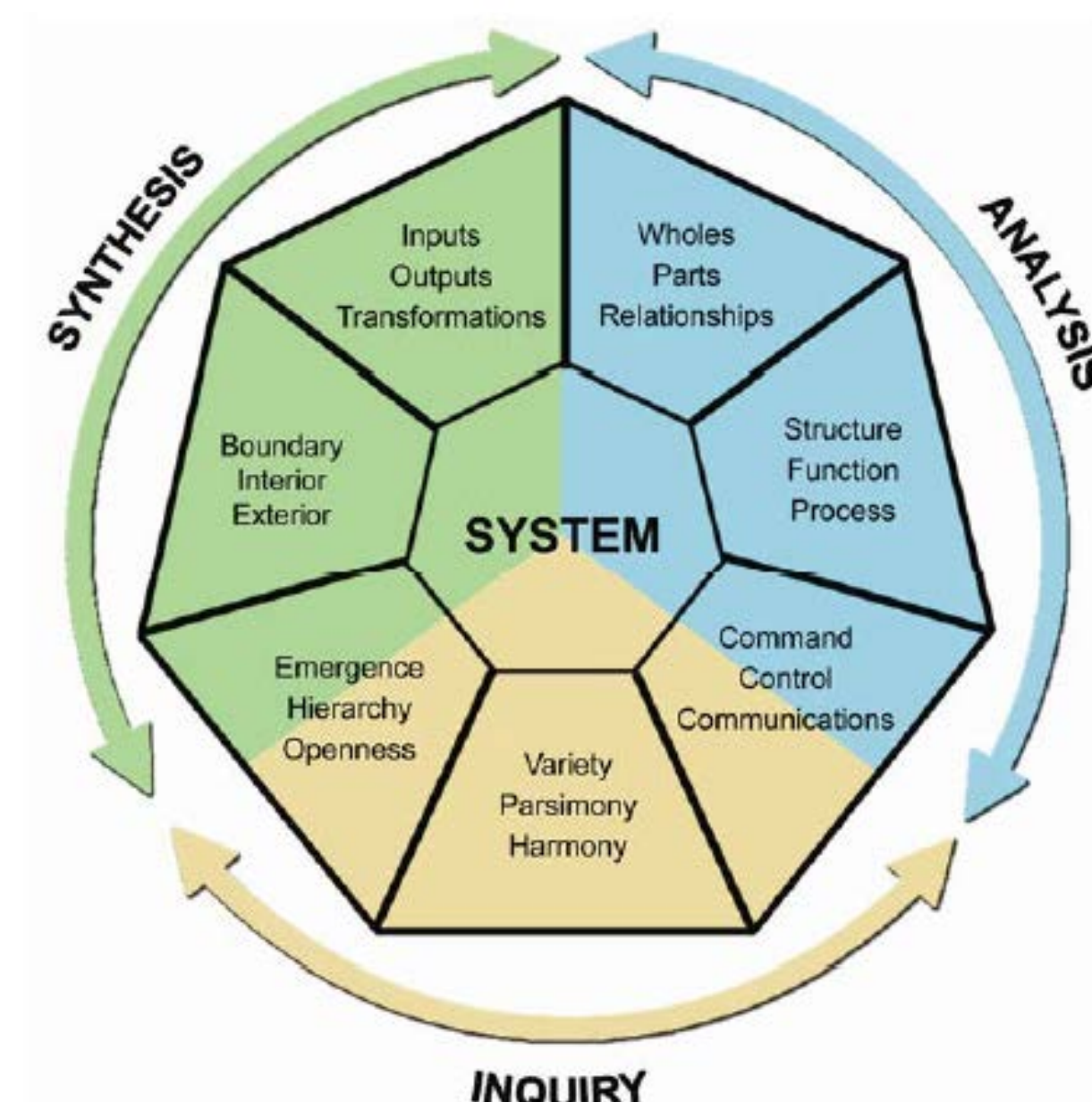
- independency
- interdependency
- distributed
- emergent behavior
- evolutionary development

### SoS Engineering Elements



Dahmann, J., Baldwin, K. J., & Regovich, G. (2010). Systems of Systems and Net-Centric Enterprise Systems. Retrieved from [http://www.mitre.org/sites/default/files/pdf/09\\_0497.pdf](http://www.mitre.org/sites/default/files/pdf/09_0497.pdf)

### System Thinking Framework



Edson, R. (2008). *Systems thinking, applied. A primer*. Retrieved from [http://www.anser.org/docs/systems\\_thinking\\_applied.pdf](http://www.anser.org/docs/systems_thinking_applied.pdf)

## Application SoSE to Networked Research EcoSystem (NetRES)

### NetRES SoS Architecture - Organizational View

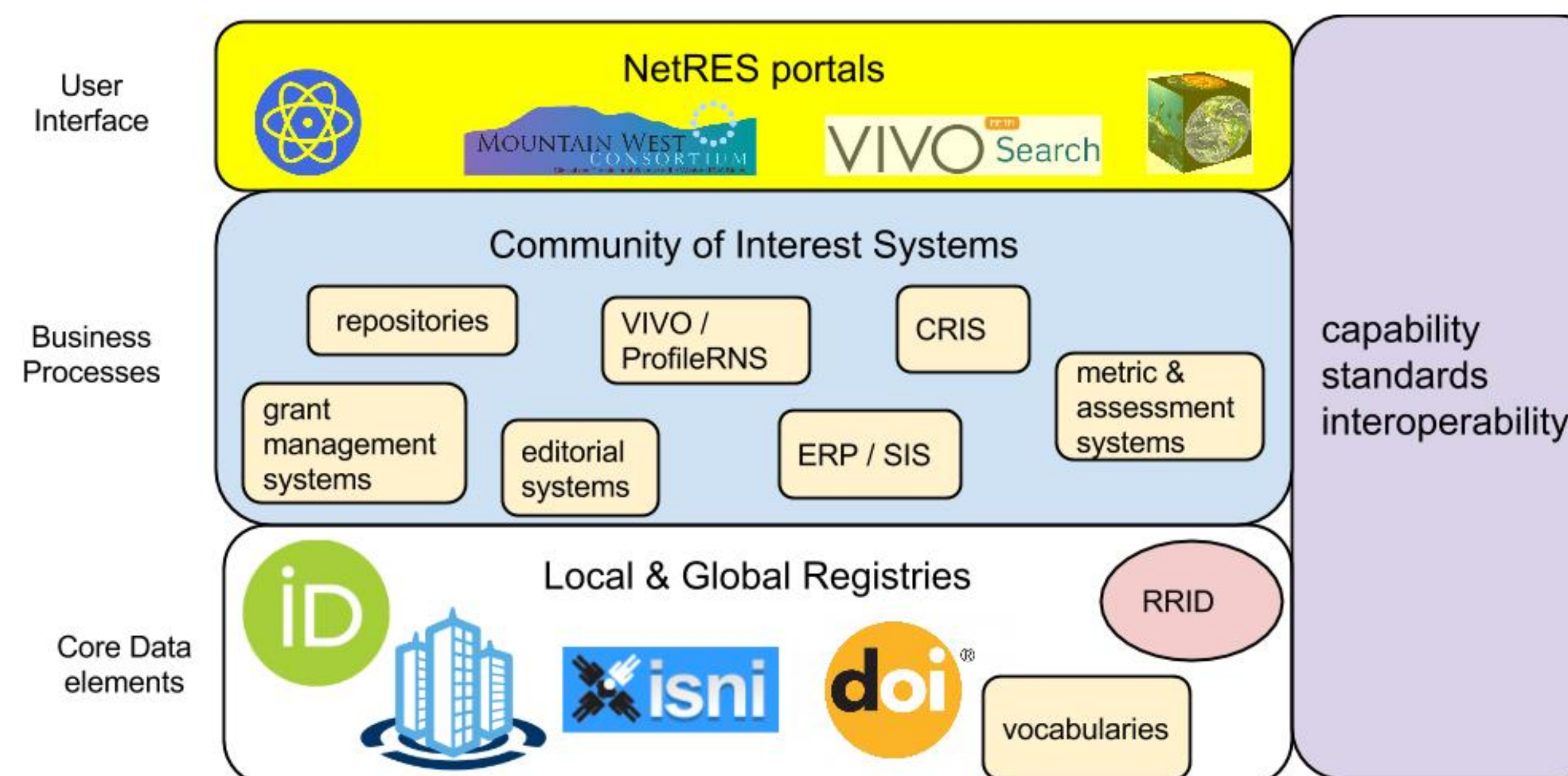
Extended roles for working groups to build sustainable Networked Research Ecosystem:

**User experience:** engaging experts from various domains to identify SoS capability and user operational requirements.

**Outreach:** strength collaboration among NetRES stakeholders, improve interoperability and LOD capability across NetRES constituent systems.

**Architecture & Technology:** coordinating documenting, design and implementation of NetRES architecture changes; developing & implementing of LOD technology and standard for connecting and sharing data.

### NetRES SoS Architecture – System View

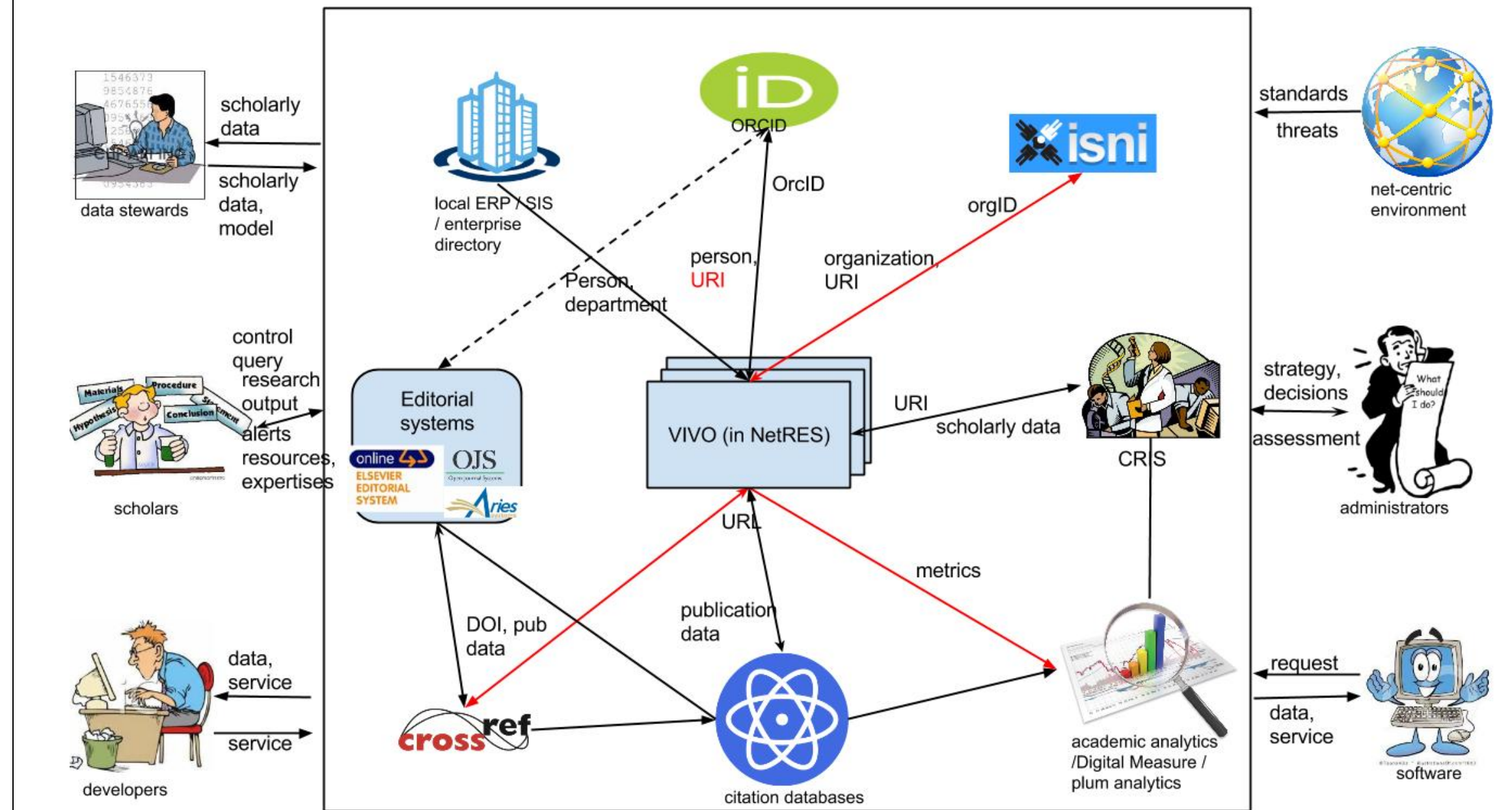


### Conclusions

System of Systems Engineering principles and methodology can be applied to Network Research Ecosystem (NetRES) and guide the evolutionary development of NetRES and its constituent systems, including VIVO and its integration with the other constituent systems within NetRES. The two approaches to build evolvable, affordable, and sustainable NetRES include:

- strengthening partnership across diverse communities
- developing SoS architecture over time

### Context of Publishing in NetRES



### Scenarios

Publication with multiple authors affiliated with different institutions:

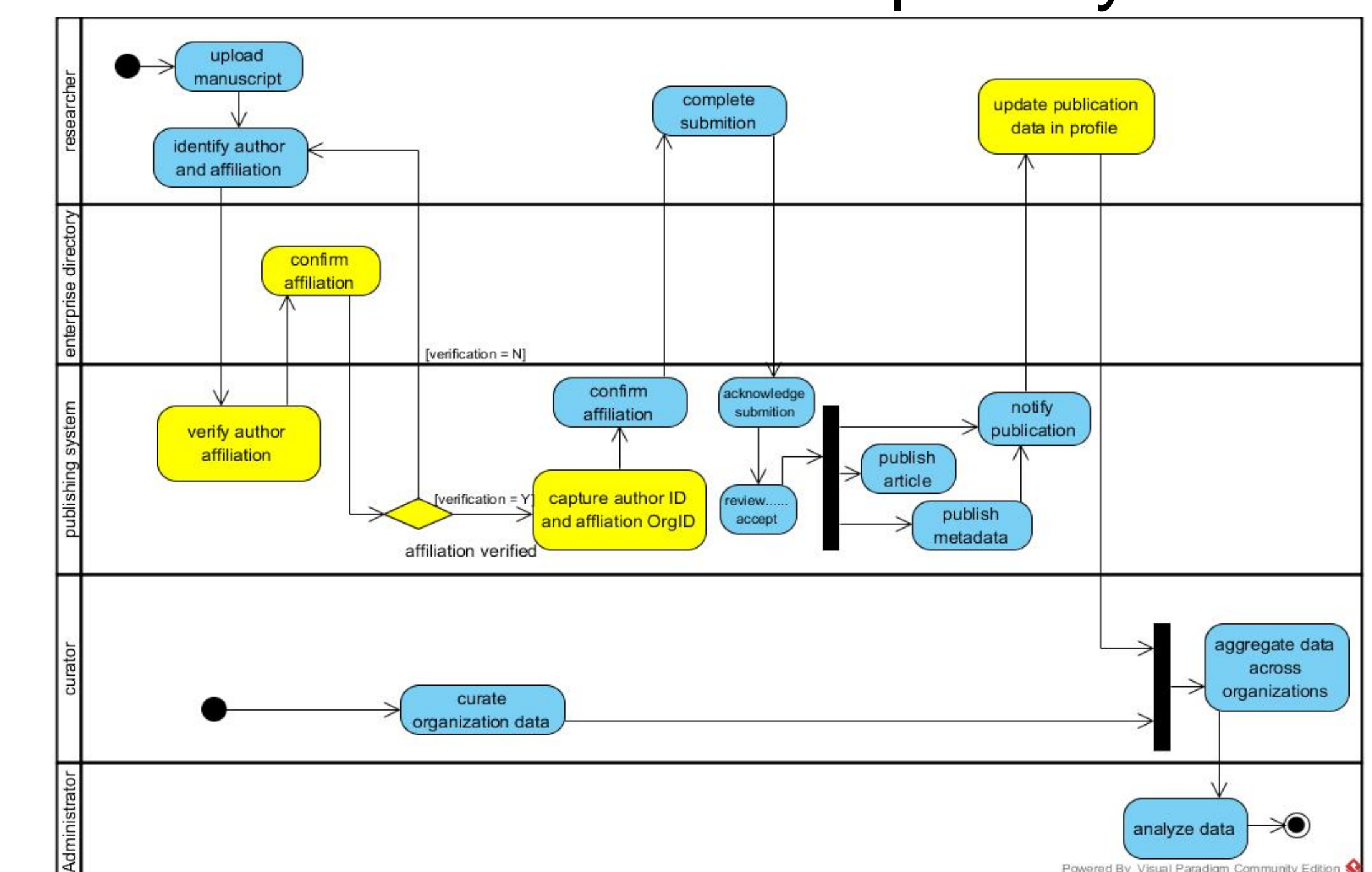
**Authors** need to submit co-author's name and affiliation to publisher,

**Authors** want to add the publication to their personal website and/or profiles.

**Users** who found the article in either VIVO sites, or Scopus, or anywhere on the internet, shall be able to link to the authors' current profile page to view more information about the author's expertise.

**Administrators** of various organizations need to aggregate and evaluate research impact.

### End-to-end capability



### SoS Integration Properties

Integrity / consistency*	ensures the validity and integrity of the data shared between systems when integrated.
Scalability*	scalable across large number of systems with various difference (e.g. data model, version, customization, policy)
minimum data redundancy	data is stored or indexed only when it is necessary, otherwise only exists in the authoritative source
system independency / manageability	individual system remains autonomous, which enables its easy evolving, affordable sustainment and substitution, and enables greater flexibility to promote better evolution of the SoS.
interoperability	automatically interpret the information exchanged meaningfully, accurately, and easily in order to produce useful results as defined by the end users of both systems
Performance*	integration activities do not require too many intermediate steps or excessive executions and data transfers
Reliability*	integration works reliably, at least system initiating communication is informed if the integration breaks down.
Availability*	requires the integration source / destination remain available.

\* Kazman, Rick., Nielsen, Claus., & Schmid, Klaus. (2013). *Understanding Patterns for System-of-Systems Integration* (CMU/SEI-2013-TR-017). Retrieved July 30, 2015, from the Software Engineering Institute, Carnegie Mellon University website: <http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=75750>