A Systems Engineer's View of SOA

Dr. Rob Cloutier
Associate Professor
School of Systems and Enterprises
Stevens Institute of Technology

April 2, 2009

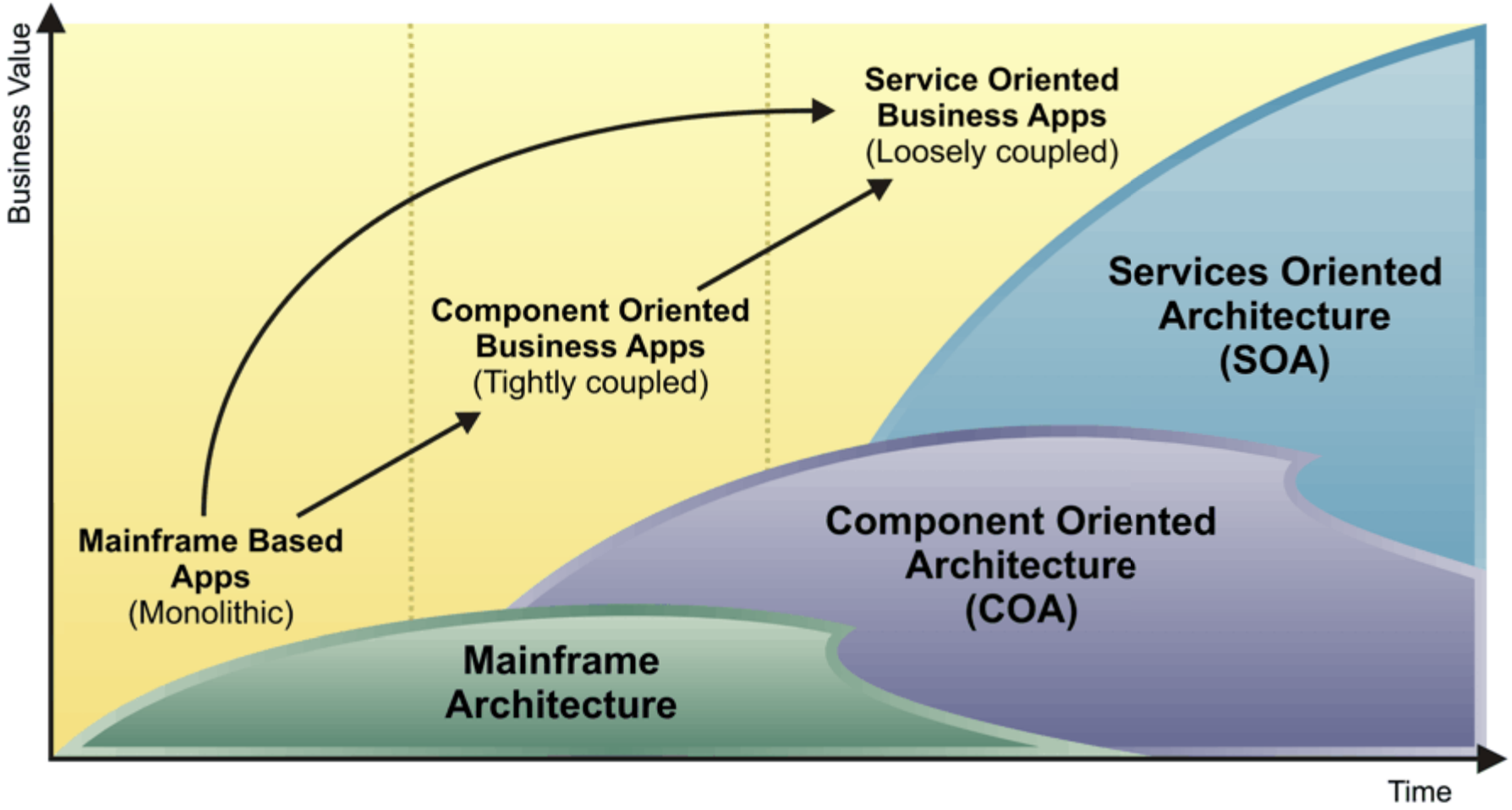
Systems Engineers engineer systems which…

… are man-made, created and **utilized to provide services** in defined environments for the benefit of users and other stakeholders. These systems may be configured with one or more of the following: hardware, software, humans, processes (e.g., review process), procedures (e.g., operator instructions), facilities, and naturally occurring entities (e.g., water, organisms, minerals).

In practice, they are **thought of as products or services**. The perception and definition of a particular system, its architecture and its system elements depend on an observer's interests and responsibilities. One person's system-of-interest can be viewed as a system element in another person's system-of-interest. Conversely, it can be viewed as being part of the environment of operation for another person's system-of-interest.

ISO/IEC 15288, page 52
Systems engineering — System life cycle processes

IS/IT has embarked on a new wave of computing referred to as Services Oriented Architecture

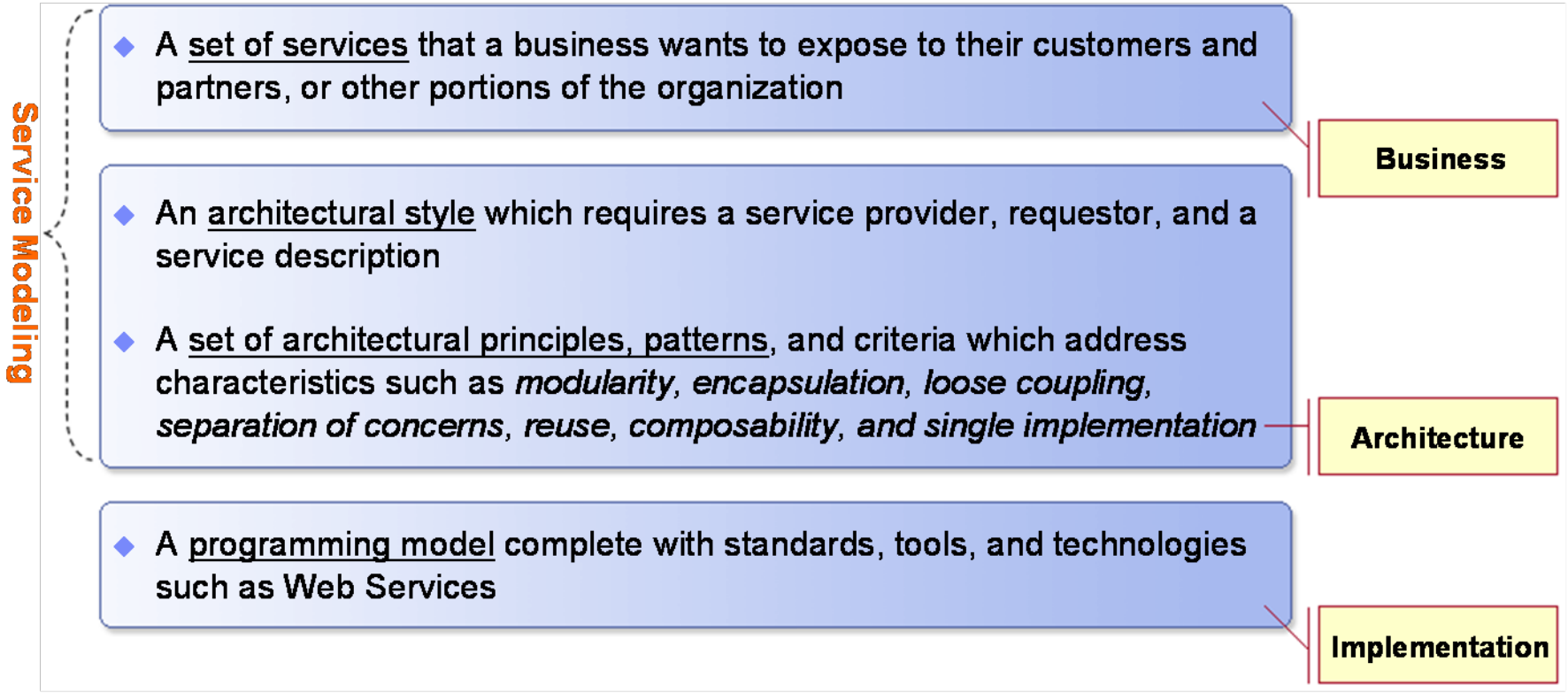


SOA Today and Tomorrow, Kerrie Holley, IBM Fellow, Worldwide CTO SOA Center of Excellence, Copyright IBM Corporation, 2005.

Business Context of SOA

"SOA in context …"

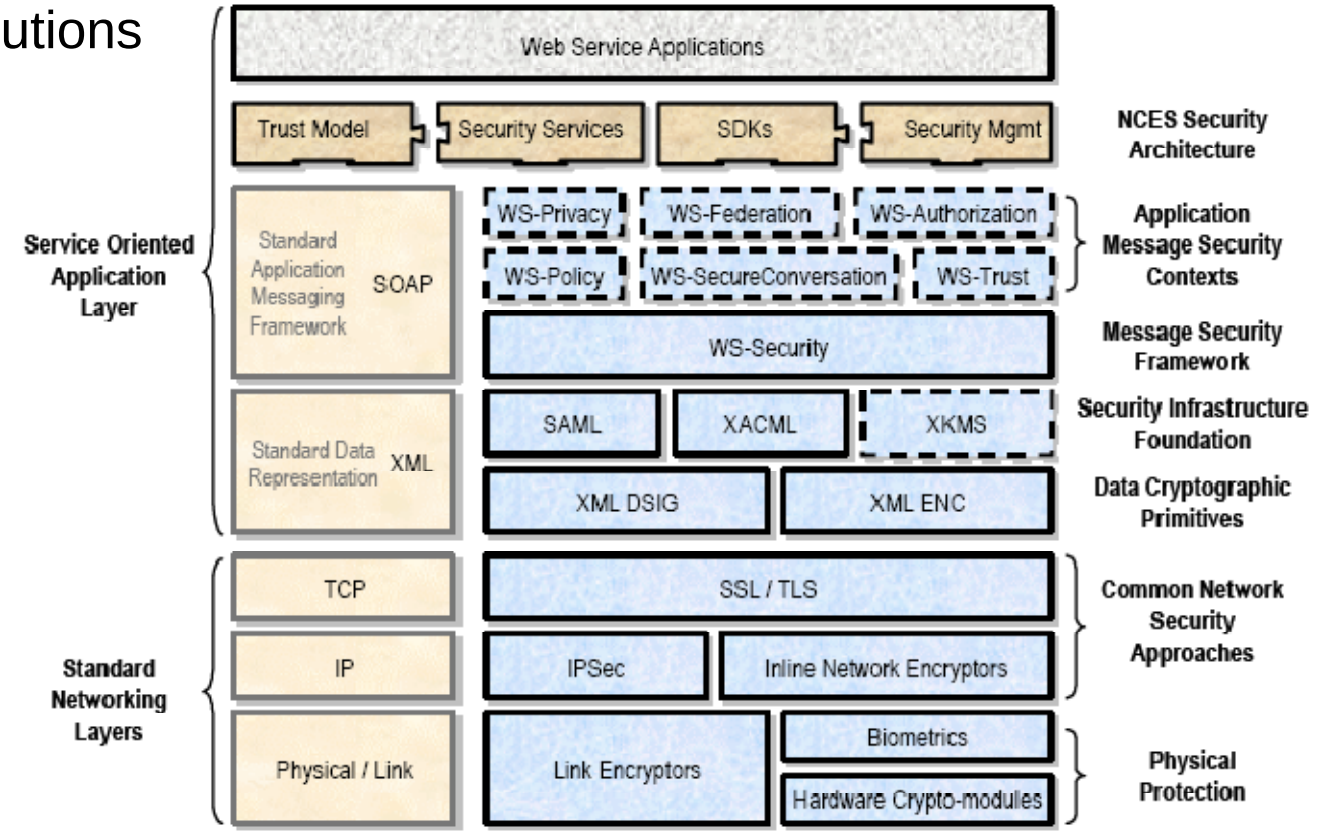
Business flexibility Through Flexible Services as the Unifying Construct



SOA Today and Tomorrow, Kerrie Holley, IBM Fellow, Worldwide CTO SOA Center of Excellence, Copyright IBM Corporation, 2005.

The IT View of SOA?

- This is the answer you normally get when you ask the question…
- A collection of implementation solutions based on software technologies

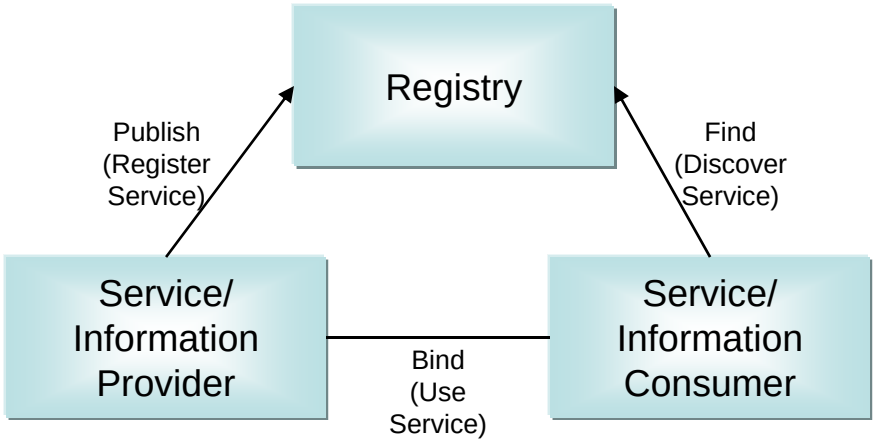


My Working Definition of SOA

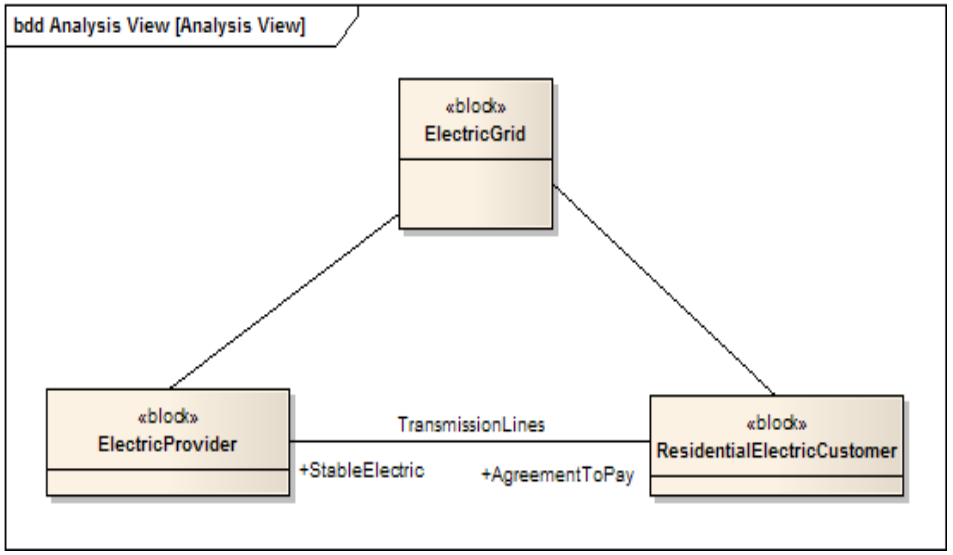
Service Oriented Architecture is a **means of organizing solutions** that promotes **reuse**, **growth** and **interoperability**. It is **not itself a solution** to domain problems, but rather an organizing and delivery paradigm that enables one to get more value from use both of capabilities which are locally "owned" and those under the control of others.

OASIS, Reference Model for Service Oriented Architecture,
OASIS committee specification 1, 2 August 2006

The Simple SOA Model



US Electricity Production and Delivery SOA Example

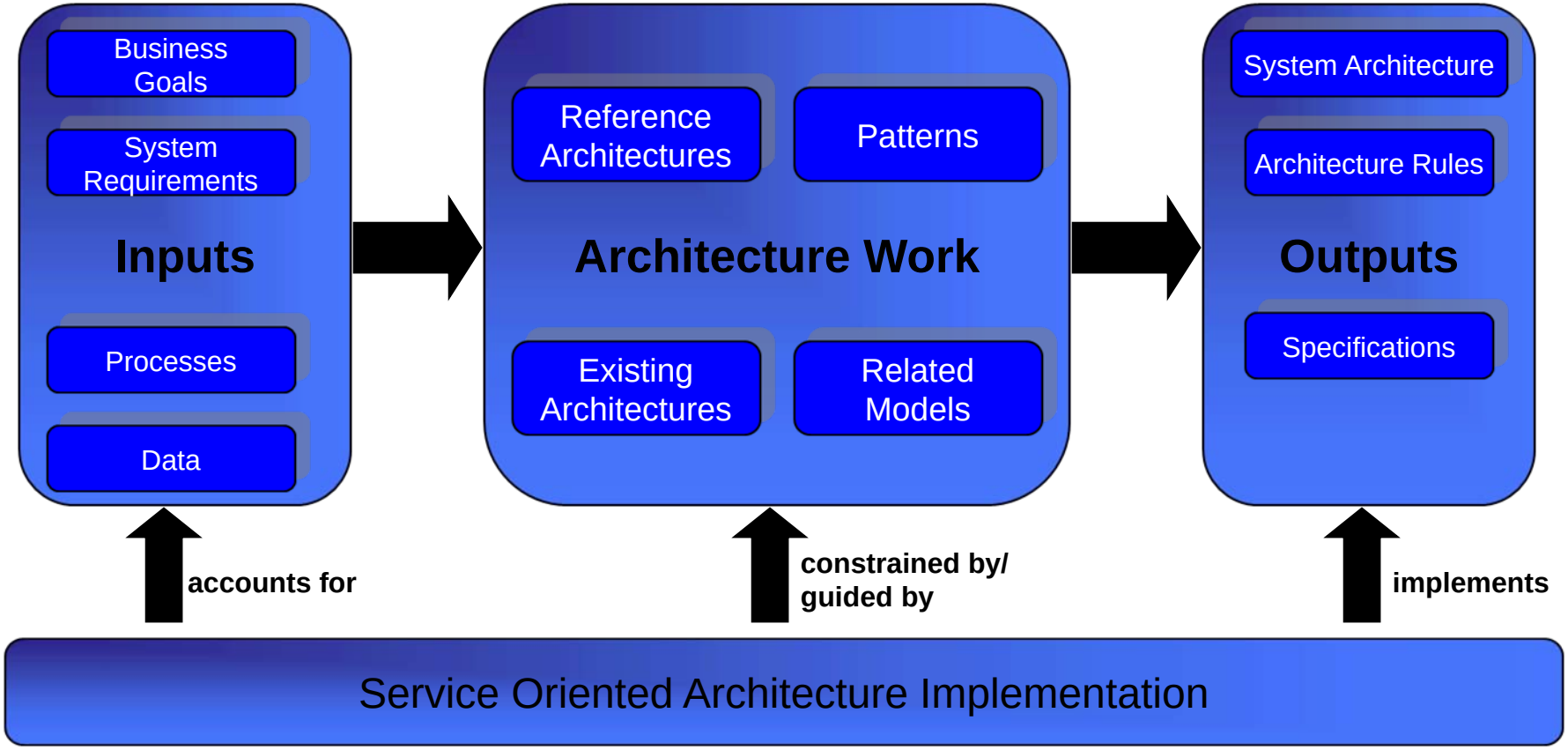


- Service Level Agreements include:
  - Consistent Electricity at constant Voltage & Wattage, under varying loads
  - Standard plugs

Some Interesting Observations

- Consumer can add new AC, Freezer, etc. without notifying electric provider
- Without the electric meter, the electric company has no idea what each consumer consumes
- There are multiple providers on the Grid
  - Electricity is not routed to you - if you are using Green Power Company, those electrons may never arrive at your home
- What happens when demand outstrips supply?
  - Borrowed from another service provider
- Implementation of service is transparent to consumer
  - Consumer does not normally care how it is produced

System Architecting an SOA System

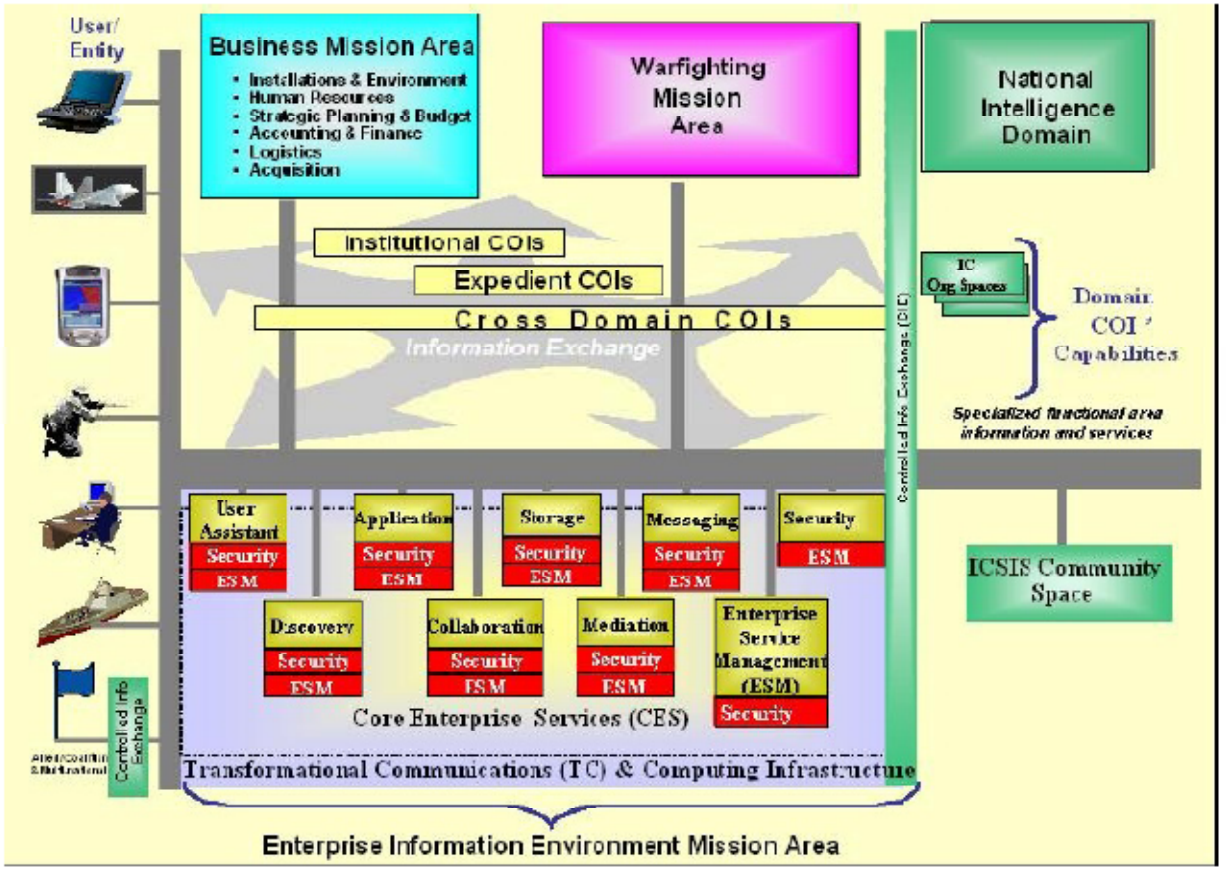


Inspired by OASIS, Reference Model for Service Oriented Architecture, OASIS committee specification 1, 2 August 2006

Implementing SOA on the GIG

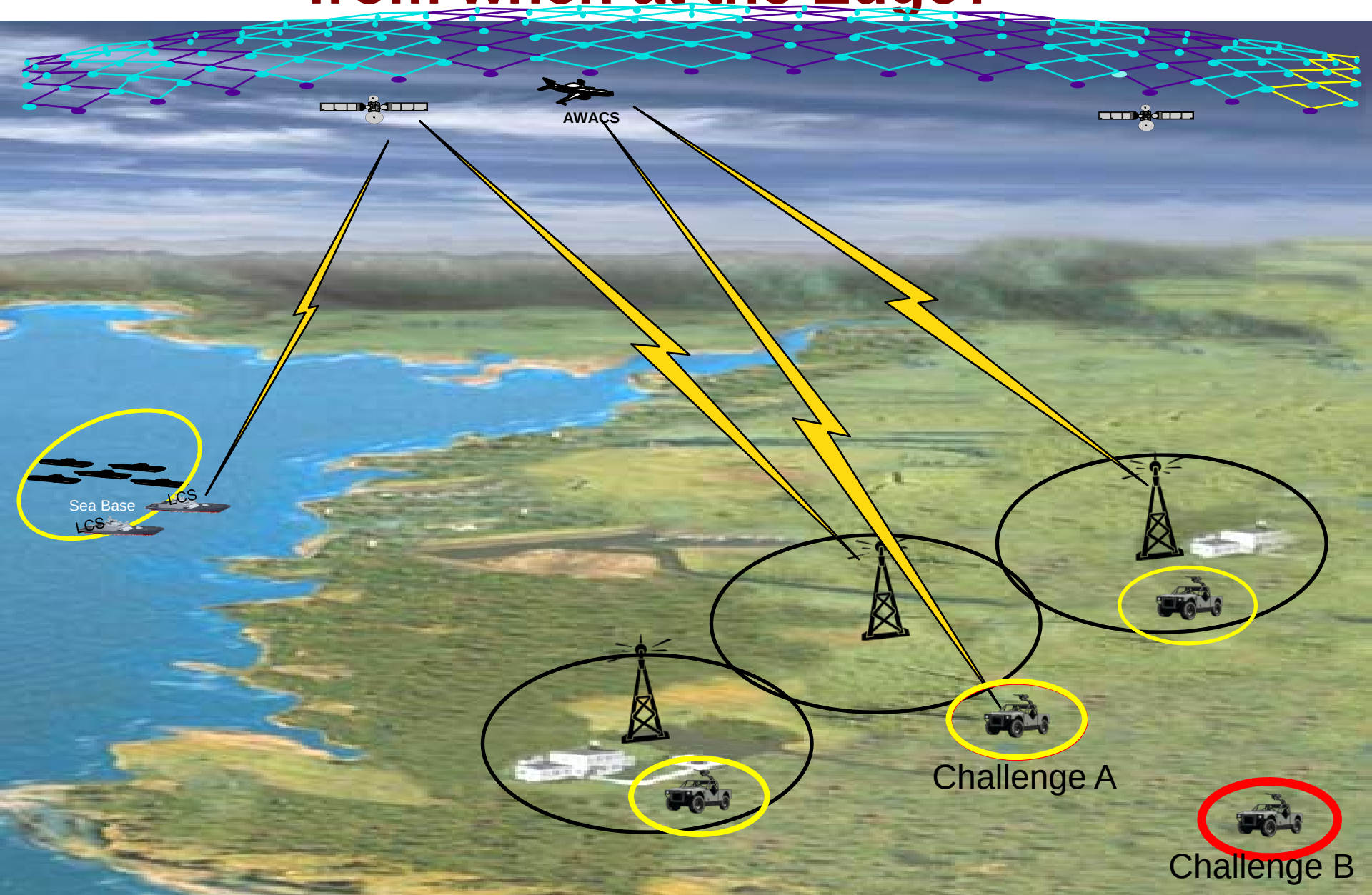
presented to The SOA Forum

GIG Enterprise Services



DoD Goes SOA, The Road Toward Net-centric Operations, Rob Vietmeyer, NCES Chief Engineer, DISA, March 2005, Presentation to SOA Forum

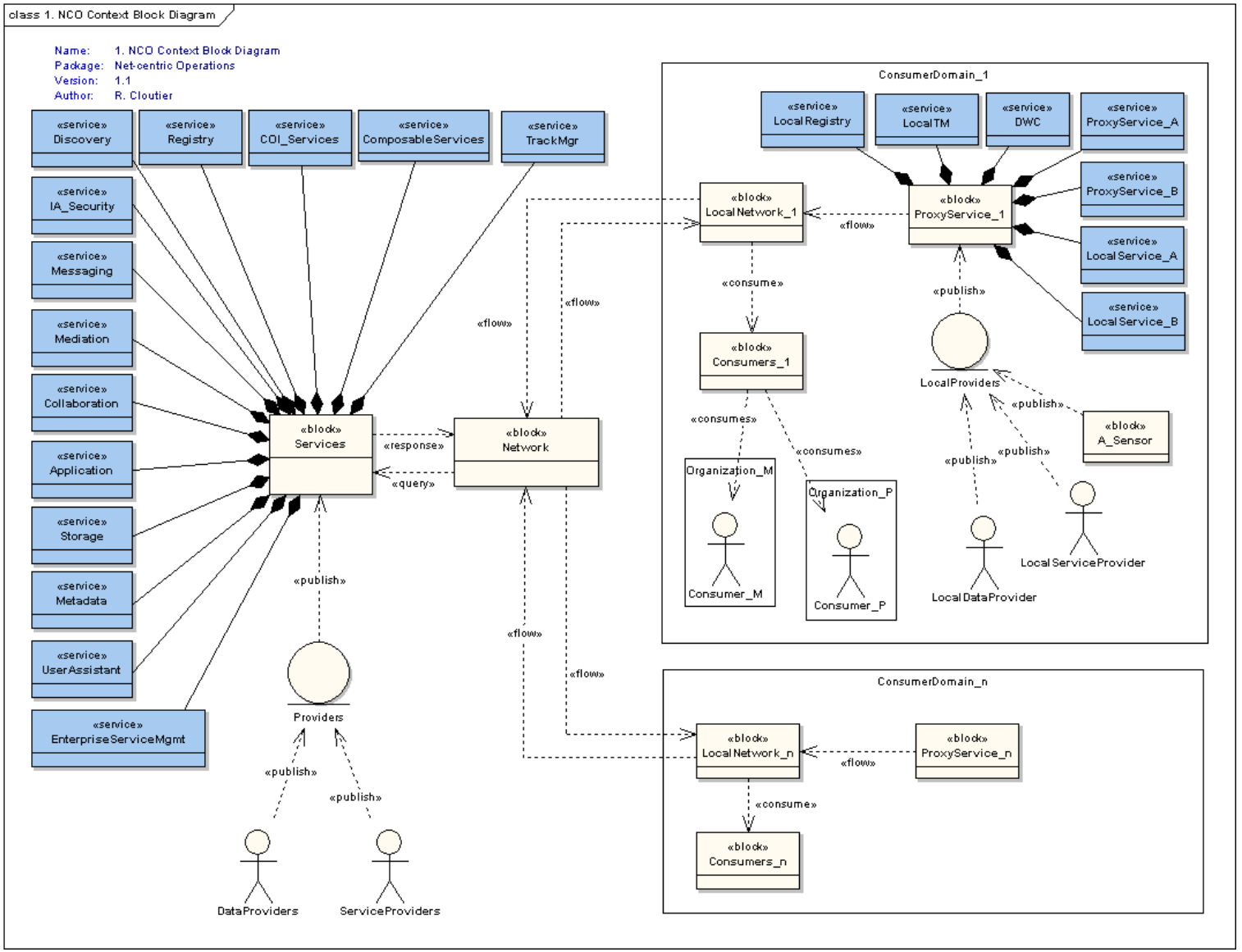
Where do the services come from when at the Edge?



Systems Engineers Must Consider…

- All possible situations
  - The fully connected user
    - Full bandwidth
  - The limited capability user
    - Limited bandwidth
    - Limited processing
    - Limited transmit power
    - Limited screen size
    - Hard to read because the vehicle is moving
    - Etc…
  - The fully disadvantaged user

Potential Net-centric SOA Pattern



This is What the System Engineer Should Care About – The BIG Stuff

- Locality of Services
- Alternate paths – what happens when something goes wrong
  - Excluded user
  - Disadvantaged user
- Performance
- Semantics
- Service Interfaces

Some Service Considerations

- Granularity: What is the level of detail/abstraction of information being exchanged?
- Latency: What is the latency of the information?
- Periodicity: What is the periodicity exchange rate for the information?
- Structure of information: What is the structure of the information being exchanged?

Closing Thoughts

- More to SOA than the implementation technologies
- Large Scale SOA is not just a Software Implementation and requires Systems Engineering discipline
  - Some of the traditional focus areas for SE still apply
  - Examples include: latency, granularity, interface definitions, etc
- Need to determine the right metrics for large scale SOA – they probably are different from what we use today