SPAWAR
Systems Center
PACIFIC

“Engineering and Acquiring C4ISR Systems Based on SOA”

Lee Zimmerman & Antonio Siordia
SPAWAR Systems Center Pacific
San Diego, CA

SPAWAR Systems Center PACIFIC

Goals for this Paper

- Better understand the migration from traditional “stove pipe” systems to systems based purely on services
- Understand the significant changes to how we specify, acquire, integrate, test and field systems as we move to services-based systems
- Discuss potential solutions for many of these challenges, using real world examples where possible

Within the Context of the Defense Department

SPAWAR Systems Center PACIFIC

The Evolution of C4ISR Systems

(1) Stove-piped System

- *Provides all if it's own functionality*
- *1-to-1 external interfaces*
- *Hosted on own hardware*

(2) Initial Net-Centric System

- *Provides all if it's own functionality*
- *Uses web services & XML to decrease # of unique interfaces*
- *Hosted on own hardware*

(3) Stove-piped SOA System

- *Provides all of own functionality as collection of services*
- *Hosted on own hardware*

(4) SOA System

- *Functionality through combination of own and shared services*
- *Hosted on hardware provided by variety of systems*

(5) Collection of Services

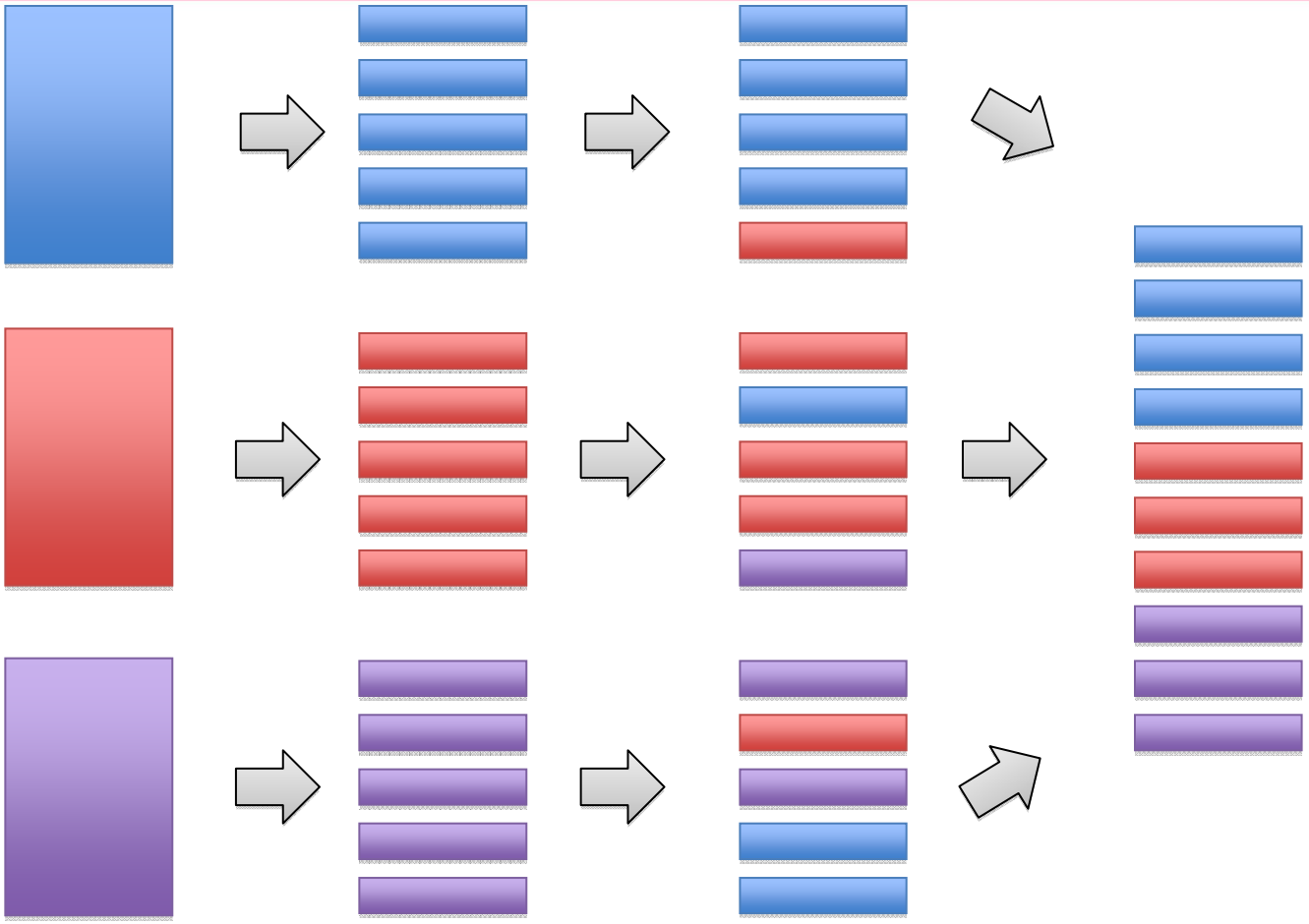
- *No longer fielding systems—fielding collection of services from multiple providers to meet mission thread requirements*
- *Hosted on virtualized hardware provided by variety of systems*

(6) Cloud Services

- *SOA Services hosted in shared data centers on cloud infrastructure*

SPAWAR Systems Center PACIFIC

Decomposing Systems Into Services



Legacy Program of Record

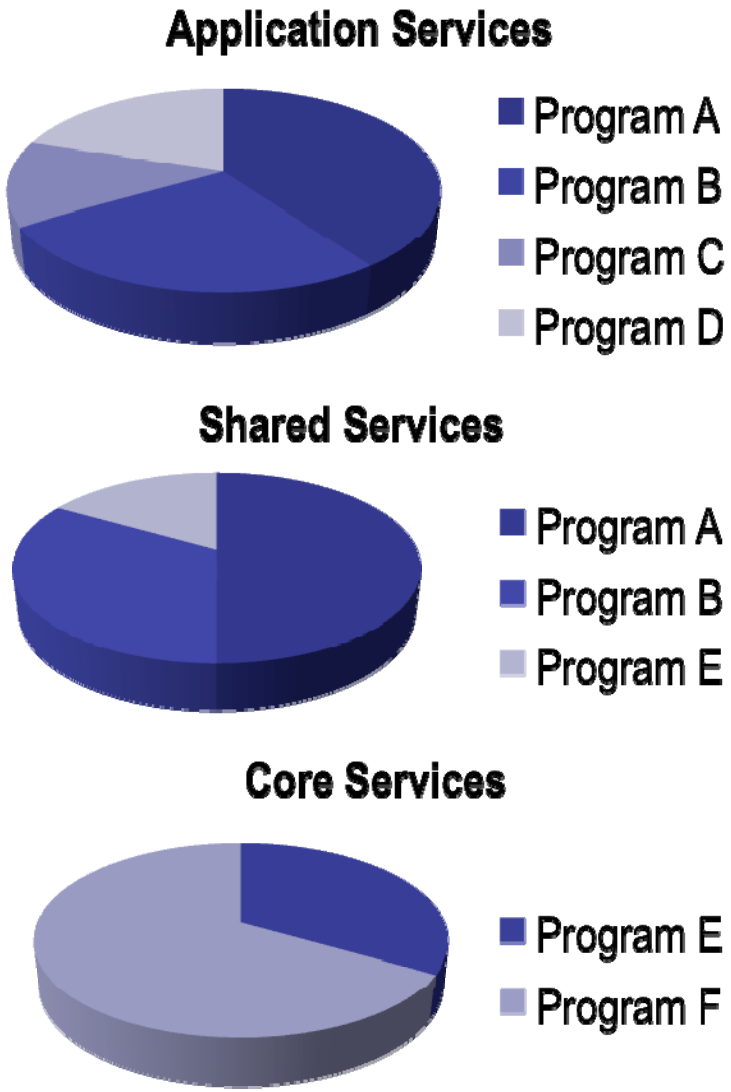
Decomposed Into Services

Starting to Use Other's Services

Collection of Services

Requirements Allocation

- Shift from system level to mission threads
- Partition solution(s) between core services, shared services and unique application services
- Importance of Service Level Agreements (SLA)
- Shared risk
- Dynamic composeability?
- How allocate funding?

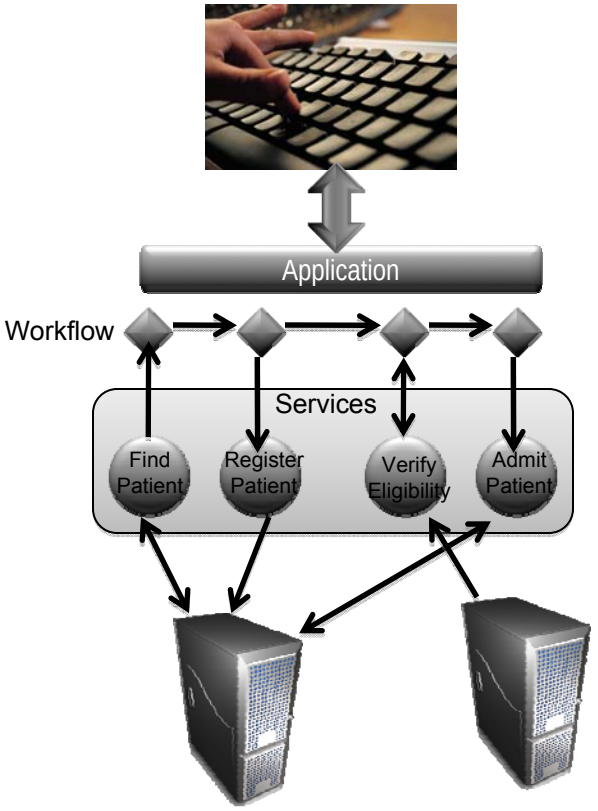


Development: What is needed?

Access to other services

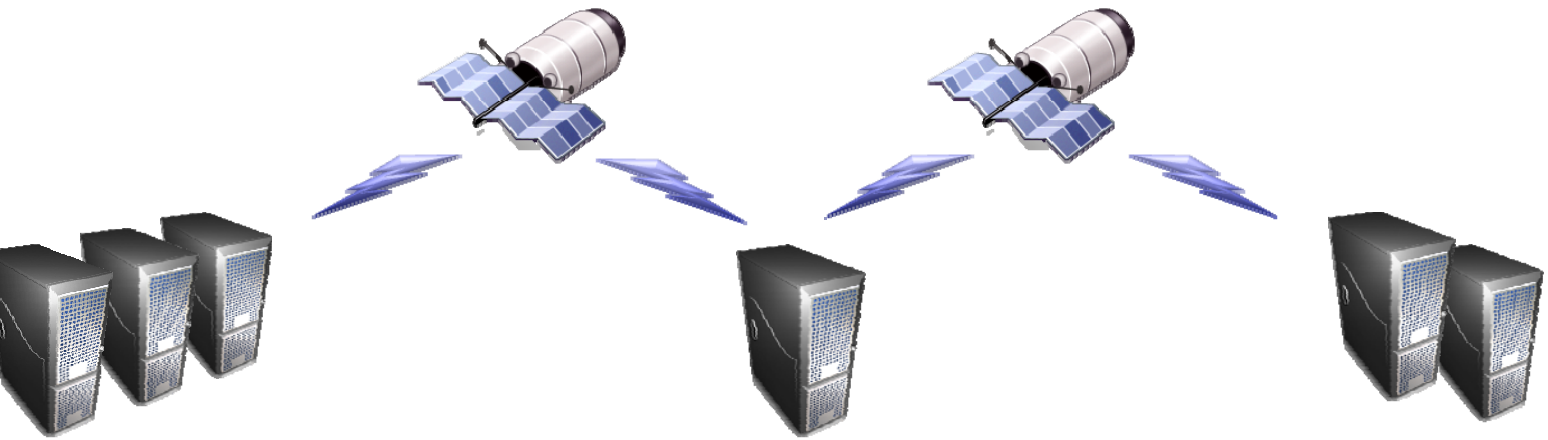
Interoperability standards

System-of-systems engineering



Testing: Concerns & Questions

- Can't test all combinations of services
- Need multiple mission threads to exercise all services - breadth and complexity of test environment
- Service vs. thread level metrics
- Baselines vs. individual service



Accreditation

Cloud based SOA system breaks almost every key tenet of current processes

- Logical system boundary
- Physical system boundary
- Crosses multiple organizations

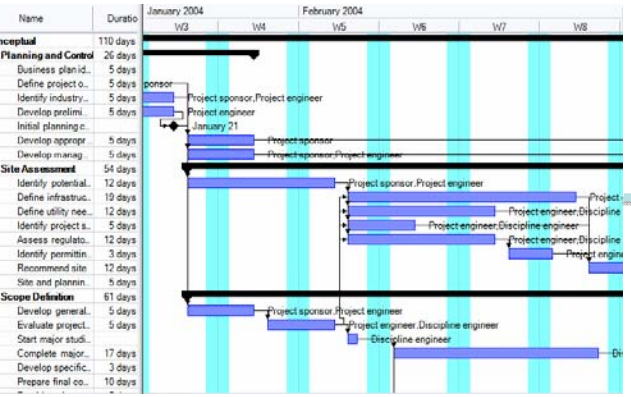
Selected Accreditation Best Practices

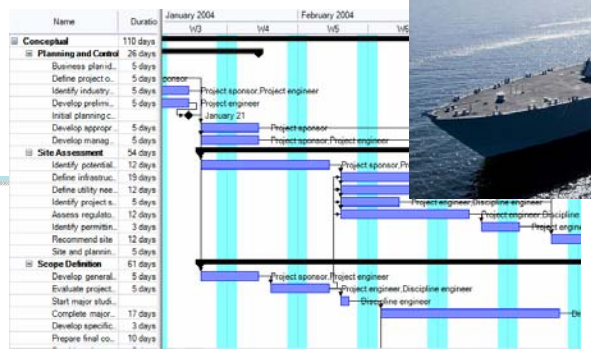
- Policy and Doctrine harmonization
- Upfront planning and architecture to identify friction points
- Better mechanisms to quickly adjudicate issues

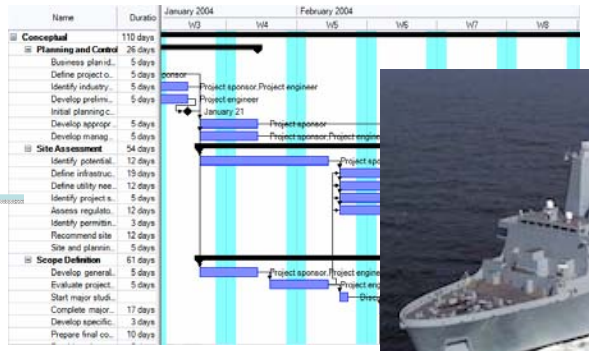
Fielding

Baseline or individual service?

Schedule alignment among services







Net-Enabled Command Capability (NECC)

Approach:

- Centralized funding and requirements, flow down to stakeholders
- Innovative test process

Pros:

- Well documented, consistent architecture, governance

Cons:

- Significant up front costs before see operational results
- Perceived loss of control by Services

Outcome:

- Program cancellation

Global Command and Control System – Integrated Imagery and Intelligence (GCCS-I3)

Approach:

- Component programs retained funding
- Common fielding schedule & hardware
- Rigid interoperability standards
- Shared user environment

Pros:

- Parallel development by small teams
- Legacy programs evolve at own pace

Cons:

- Effort to align multiple component programs

Outcome:

- Success!

Consolidated Afloat Networks and Enterprise Services (CANES)

Approach:

- Centralized Funding for hardware, decentralized funding for software
- Cross-POR governance - common services, development cycle, testing, hardware allocation

Pros:

- Evolve legacy PORs in parallel with emphasis on shared services
- Savings from common, virtualized compute infrastructure

Cons:

- Some Loss of POR autonomy

Outcome:

- TBD

Summary

Must develop acquisition and engineering processes that allow us to:

- Define requirements in terms of mission threads and services
- Allocate requirements across multiple organizations for implementation of services
- Allocate resources for sustainment of services
- Test complex collections of services
- Accredit “systems” composed of collections of services
- Field and sustain “systems’ composed of interdependent services

Way Ahead

- Continue to develop and capture best practices from efforts like GCCS-I3, CANES, C2 RPC, NECC follow-on,…
- Leverage SPAWAR Enterprise Engineering and Certification (E2C) lab environment to develop engineering, test and fielding approaches for services-based systems
- Participate in efforts to reform Defense Acquisition such as Section 804 of the U.S. DoD Defense Authorization Act, U.S. Federal Cloud Computing initiative, UCORE, …

Contact Information

Lee Zimmerman

- lee.zimmerman@navy.mil / +1-619-553-4046

Antonio Siordia

- antonio.siordia@navy.mil / +1-619-553-5601

Mailing Address:

Lee Zimmerman / Antonio Siordia
Commanding Officer
SPAWAR Systems Center Pacific
53560 Hull Street
San Diego, CA 92152

http://www.spawar.navy.mil