Where Our National Security Begins...
Software Framework Integration vs. Software Development

NRO ASP Industry Advisory Working Group

Keith Barber & Nick Buck
Software Integration vs Dev Implementation Team

Pam Arya
Keith Barber
Nick Buck
Animesh Gupta
Alex Fox
Erik Grant
Ben Avicolli
John Hays
Doug Harts
John Farrell
Ray Falcione
Michelle Combs

Cross Functional Mix from Large, Medium, and Small Businesses
Topical Background

NASP IAWG report published in 2015

Challenge identified: evolving roles
- Software developer vs COTS integrator
- Framework vs segment integrator

Industry recommendations
- Define roles, distinguish between
- Cross-walk government thinking with industry trends

Driving Questions:
Is there a difference between software development & integration?
If so, how is NRO industry base affected?
Alternative Segmentation & Integration Models*

- Segmentation model must precede OCI model
- Key OCI Issues:
  - Who writes requirements?
  - Who makes make-buy decisions?
- Mission-specific OCI ok if consistently applied

*For illustrative purposes only. Not intended to reflect govt architecture

Striking a Balance

ASP
Industry: bigger investment
Competitive field: smaller
Prime Contractor Integration

ASP-NASP
Industry: Medium investment
Competitive field: broader, by msn
Mostly Industry integration

ASP-Framework-NASP
Industry: Focused investment
Competitive field: broadest
Govt-Industry integration

ISP: CME
Layer 4 Computing
Layer 3 Networks
Layer 2 Fiber/Cable
Layer 1 Facilities

ISP: CME
Layer 4 Computing
Layer 3 Networks
Layer 2 Fiber/Cable
Layer 1 Facilities

Composable ASP
Mission Unique Applications
Mission Framework
Common NASP Services

Common NASP Services
Mission Framework
Mission Unique Applications

Common NASP Services
Mission Framework
Mission Unique Applications

Common NASP Services
Mission Framework
Mission Unique Applications
Software Integration vs Development Related Issues

- NPS wargame: promoting lock-in vs promoting open competitive ecosystem
  - Finding: software integrators have power to create lock in from the beginning
  - Observation: GOTS can be a form of lock in. Both COTS and GOTS susceptible.
  - Finding: specific steps must be taken to minimize lock in

- Closely related to COTS vs GOTS business model
  - Pure GOTS development using OSS module integration
  - Pure COTS integration using out of the box configurations (incl OSS modules)
  - Hybrid: GOTS “extending” COTS via API/SDK beyond out of box configurations

- Source code vs Executable code
  - Government proscribed deliverables & requirements must address COTS/GOTS
  - Government assumption/preference to GOTS because COTS requires licensing?
Is it Software Integration or Development? Criteria

Criterion: Configuration vs Coding?
- Coding to transform and normalize data
- Coding for functionality
- Configuration of software functionality in executable/run time

Criterion: “Wrapping”? A common practice. In terms of integration vs development:
- Am I abstracting/concealing vendor IP?
- Am I removing/reducing visibility of sub-components?
- Is the resulting baseline CM controlled at the bundled level or at the component level?
- Is wrapper code maintained separate and distinct from components or as an integrated unit?

Criterion: Testing approach?
- Single item (dev) vs multiple item (int) e.g. unit test vs Multi-segment testing
- Performance testing vs functional testing vs load testing
Software Integration vs Development Roles
Discussion

What are the tradeoffs?
- Requirements: individual products may not meet requirements or fit architecture even when config’d
- Support tail: COTS & GOTS wrap components; COTS amortizes maintenance cost
- Integration risk: government assumes more risk when wrapping not permitted
- Security: component level control brings higher viz to code security but may result in exploits; bundled software may enhance code security and reduce inter-component exploits
- Testing differences between an integrated software baseline vs a collection of components

Role of Source code vs Executable code

“Web Development”…does it redefine or reframe the questions?

RECOMMENDED ACTION: assemble scenarios showing the spectrum
- No COTS available to meet requirements
- COTS ubiquitous to meet requirements
- Hybrid mix of COTS/GOTS to meet requirements
Reference Data from Report & Reccos

NRO ASP Industry Advisory Working Group

March 2015
As the NASP gets instantiated and providers are defined – the questions will be:
- ASP-ISP: What is the NASP Relationship to Services from CME, C2S?
- Intra-ASP: What is the tie between Frameworks and the applications that ride on them?

Contrasting integration models
- Software system (M2C2) riding on an ISP (CME/C2S)
- “Composed” systems on ASP (S2P) core services on ISP

Impacts to consider
- NRO industry base composition
- Size and number of opportunities
- Roles and expertise of “Mission SMEs”

Is the issue really OCI or is it actually technical objectivity?
What unique business challenges exist for NASP that are influenced by OCI?

**NASP IAWG Action Team Question:**
*What are the industry base and business model implications of an OCI between frameworks/integration and development/products?*
Segmentation ≠ OCI

“START WITH THE END IN MIND”

- Segmentation based on scope, complexity, & program objectives
- Balance between # of opportunities & integration complexity
- Emergence of multiple “integrator” roles: Program, S/W, System
- Complexity of program-by-program OCI is risk to industry base

*For illustrative purposes only. Not intended to reflect govt architecture*
• Govt interprets market research and industry exposure as “unfair access”
  – Dialog has all but shut down as early as RFI stage
  – Technology discussions do not breach threshold of Sensitive/Proprietary discussions in market research phase, pre-Draft RFP

• OCI has changed/shifted the NRO Industry base and capacity
  – Use of OCI to change supplier base (versus requirements a la C2S)
  – Bounded competitive environment may have been goal; inconsistency & delay are reality

• OCI picture in ASP-ISP world differs from vertically integrated system world
  – OCI critical to government development programs
  – Different landscape for non-developmental and COTS integration programs

• ISP-ASP OCI picture differs from ASP-ASP picture
  – ISP-ASP involves performance measurement of applications, unequal access to info
  – Framework Integrator to ASP OCI eliminated if government decides on products/specs

OCI policy and implementation should be reviewed in light of move from legacy “development” model to the new “integration” model
Recommendations (1 of 2)

• Recognize difference between software development vs COTS integration
  – Development: built uniquely to government requirements (Higher OCI potential)
  – Non-developmental: built to industry-wide requirements (Lower OCI potential)

• Recognize new integration model = new roles & OCI picture
  – Multiple versions of “integrator”: Program, System, Framework, Application
  – Government as “Program Integrator”: direct schedules, content, and product selection
  – Fundamental issue: who decides vs. who implements in a non-developmental program

• Determine/Evaluate ASP segmentation options & review ecosystem roles
  – This should be a long-term decision to provide continuity and predictability
  – SETA vs SI, who owns architecture, who creates/owns design?
Where Our National Security Begins...