Software Engineering Institute | Carnegie Mellon

Disciplined Reuse Could Come to the Rescue for Army Common Operating Environment

Reed Little
Distinguished Principal Engineer

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM-0001063

Software Engineering Institute (SEI)

Department of Defense R&D Laboratory (FFRDC)

Created in 1984

Under contract to Carnegie Mellon University

Offices in Pittsburgh, PA; Washington, DC; and Frankfurt, Germany

SEI Mission: advance software and related disciplines to ensure the development and operation of systems with predictable and improved cost, schedule, and quality.

Topics

Common Operating Environment

Computing Environment

Disciplined, Systematic Reuse

Essential Activities

Making It Happen

Wrap-Up

COE Goals

Reduce the development timeline

Lower development cost

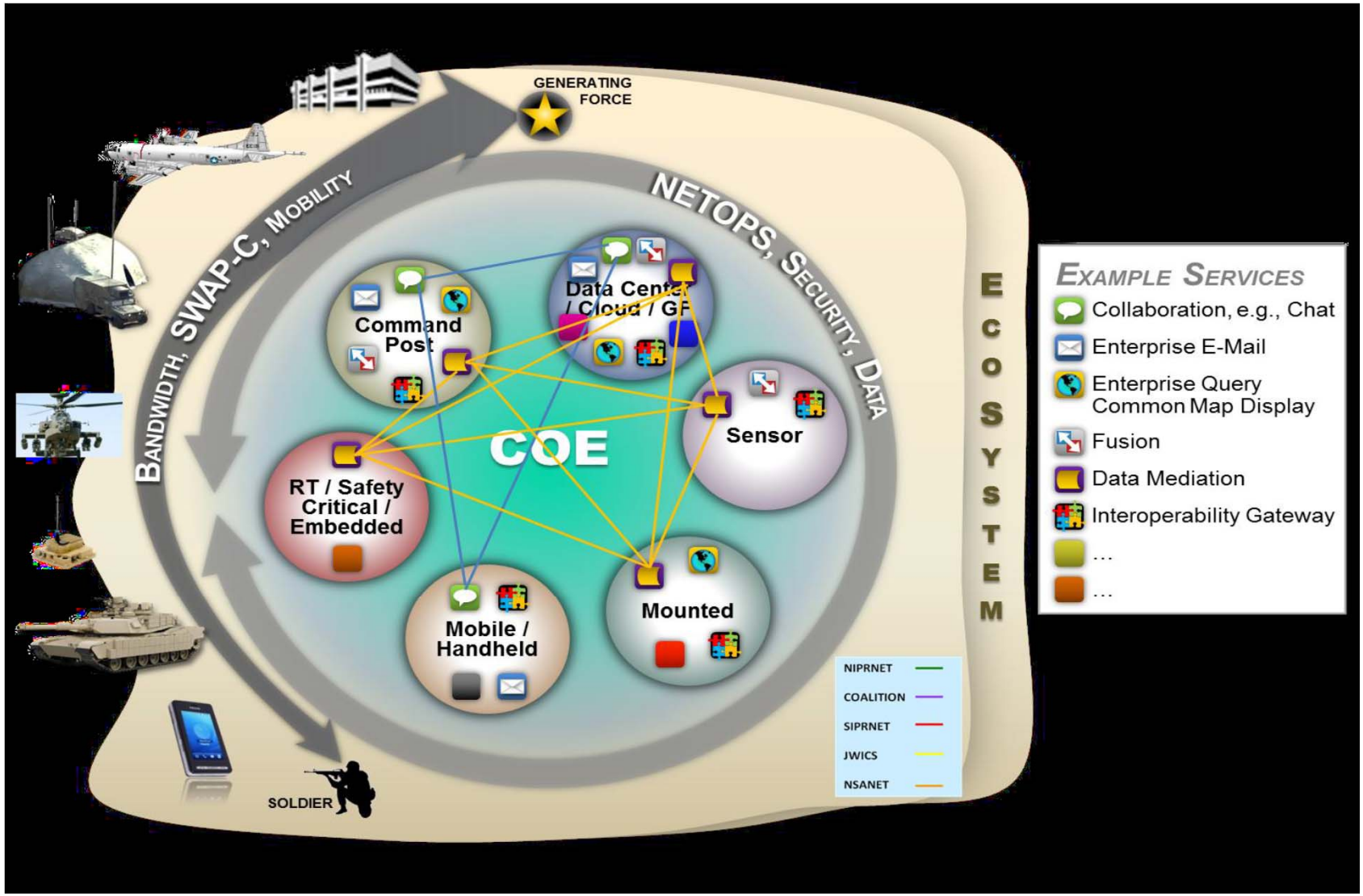
Shorten the time required to integrate and certify systems

Produce high quality systems

Faster, Better, Cheaper

ALL THREE !

COE Goals Snapshot



COE Objectives

Approved set of computing technologies and standards

enables secure and interoperable applications

rapidly developed via and executed on

a variety of Computing Environments.

COE Approach

Standardize end-user environments

Standardize software development kits

Establish streamlined enterprise software processes that rely on common pre-certifiable, reusable software components

Develop deployment strategies that give users direct access to new capability

COE Tenets

Move from hardware-centric to software-centric development

- focus on the applications and services
- utilize “off the shelf” as first option
- abstract software from hardware

Implement Service-based Architecture Approach

Execute Phased Implementation

Establish Common Frameworks and Shared Infrastructures across Computing Environments

Execute Unity of Effort across all deployment phases

COE Value Proposition

The primary COE Value Proposition is that if implemented across Army systems it will greatly increase interoperability, agility, security, safety and operational relevancy and effectiveness; and decrease time for development and delivery to the field, certification, and overall costs. Specifically, it will enable:

- Increased Capability Agility
- Reduced Life Cycle Costs through standardized applications and Unity of Effort
- Flexible Infrastructure to Evolve to Rapidly Emerging Standards
- Enhanced Cyber Protection

ALL THREE !

Computing Environment

Baselined on a common foundation (hardware and software) to facilitate reuse of common components

Designed to interoperate with other CEs, thus forming the COE

Interfaces between CEs will be enabled through the establishment of Control Points

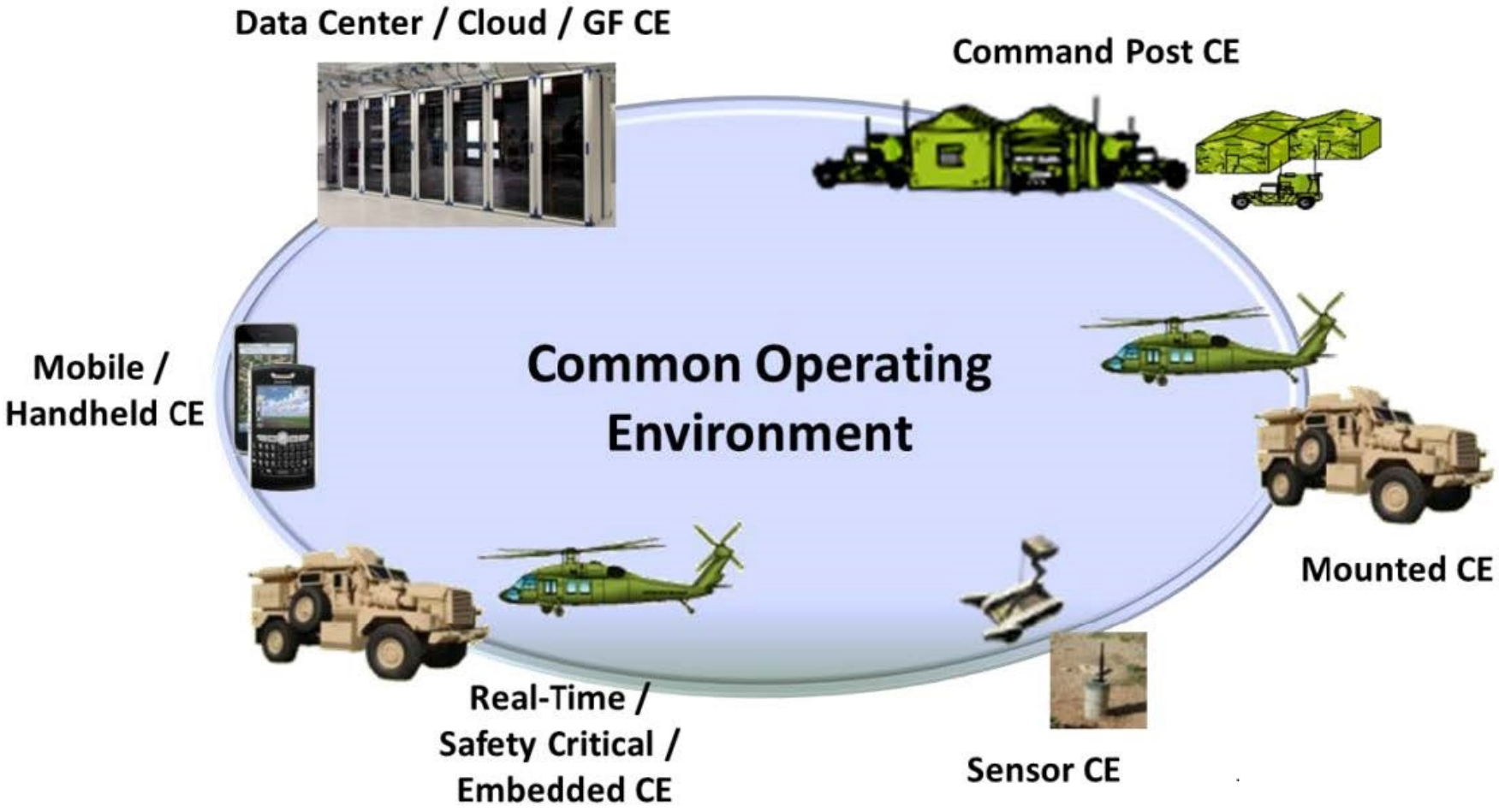
- tightly controlled technical specifications that act as the blueprint for how data will be exchanged between CEs

Alignment of Army programs into six CEs based on:

- mission and environment (size, weight, power, and bandwidth) limitations/constraints

Implementation will be in a phased approach expected to be executed over the next several years

Initial Set of Computing Environments



Inside a Computing Environment

"A minimum standard configuration that will support the Army's ability to produce and deploy high quality applications quickly while reducing the complexities of configuration, support, and training." - U.S. Army CIO/G6

CE has two major parts

- system production capability
- fielded system runtime support

Computing Environment Challenges

Some of the many challenges that must be addressed

- transition from a systems focus to a software applications focus
- effectively reuse software and architectures
- created a climate where Programs of Record can integrate applications built by others
- align system requirements to the various CEs
- establish a process to sunset PORs that are not required; and building the correct cost models
- deal with rice bowls
- develop and evolve the CEs's reusable assets

Reuse to the rescue

Computing Environment & Reuse - BLUF

CE should exemplify the concept of disciplined, strategic, planned reuse

CE concept is about more than a new technology

- new way of doing one's software business

Essential Activities

- Reusable Asset Development
- System Development
- Management

Strategic Reuse is Needed for Business Benefits

- Business + Technical Strategies = Strategic Reuse

What A CE Isn't

Fortuitous small-grained reuse

Single-system development with reuse

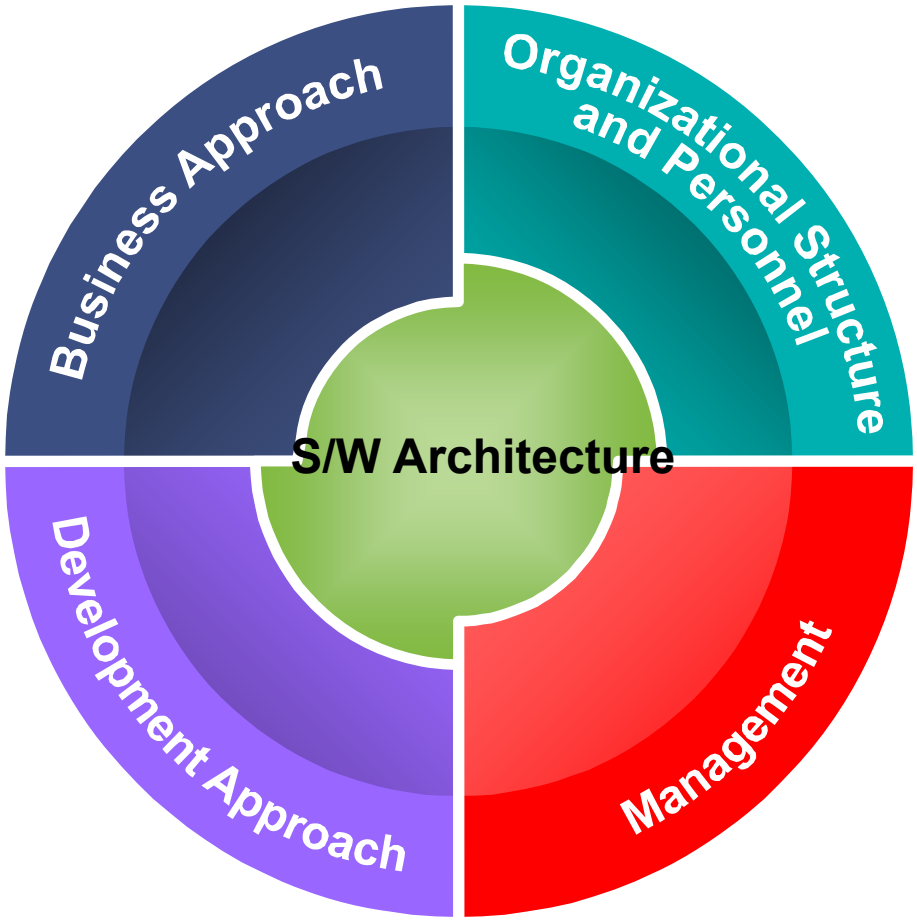
Just component-based or service-based development

Just a configurable s/w architecture

Releases and versions of single products

Just a set of technical standards

Changes Will Be Needed



Successful Strategic Reuse With A CE

Strong itch within the Army for faster, better, cheaper

Management commitment (CE developer, CE user, up the Army command structure)

Acknowledgement that CE has its own life-cycle within Army

Domain experience in CE functional areas

Process discipline

Good S/W engineering (including software architectural excellence)

The CE Production Capability Adoption Endgame

To have an **operational CE Production Capability**

To do that, a system development organization must

- have
  - a reusable asset base
  - s/w reference architecture
  - supportive processes and organizational structures
- develop systems from that reusable asset base in a way that achieves business goals
- prepare itself to institutionalize the CE Production Capability

Classic Priorities – PMO POR POV

High level

- support currently fielded system
- get next version out
- support development of new system

Life cycle for each POR typically includes:

- needs requirements
- technical planning
- resourcing
- SOW development / award
- technical oversight (SOS, SE, SW, Test, SOS test)
- training package development/support
- deployment
- sustainment

Priorities – CE Prodution Capability POV

Life cycle for each CE would typically include:

- Requirements
- Technical planning
- Resourcing
- SOW development / award
- Technical oversight (SOS, SE, SW, Test, SOS test)
- Training package development/support
- Deployment
- Sustainment

Looks a lot like POR POV, but ....

POR For Warfighter, CE For S/W Developer

Life cycle for each CE typically includes:

- Common requirements all systems supported by CE Production Environment
- CE support-from/contribution-to COE
- Reusable assets
- Cert packs
- S/W reference architecture
- SDK and build environment
- Governance structure
- Help desk
- Off-the-shelf
- Periodic updates

More like a commercial s/w product

The CE is its own "POR"

Challenges

Achieving CE and COE goals require cultural/legal/procedural changes across:

- commands
- acquisition community
- technical community
- supplier base
- Congress

The Time Is Right

Increased, different, changing threats & needs

Reduced funding to do things the old, stove piped way

Supportive technical landscape exists

- Rapidly maturing, increasingly sophisticated software reuse technologies
- Realization of the importance of S/W architecture
- Rising recognition of the significant benefits/gains that are possible

ALL THREE !

