Designing the Infrastructure for an Enterprise IT System

William E. Novak

Patrick R.H. Place

Software Solutions Conference 2015

November 16–18, 2015

Software Engineering Institute | Carnegie Mellon University

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.

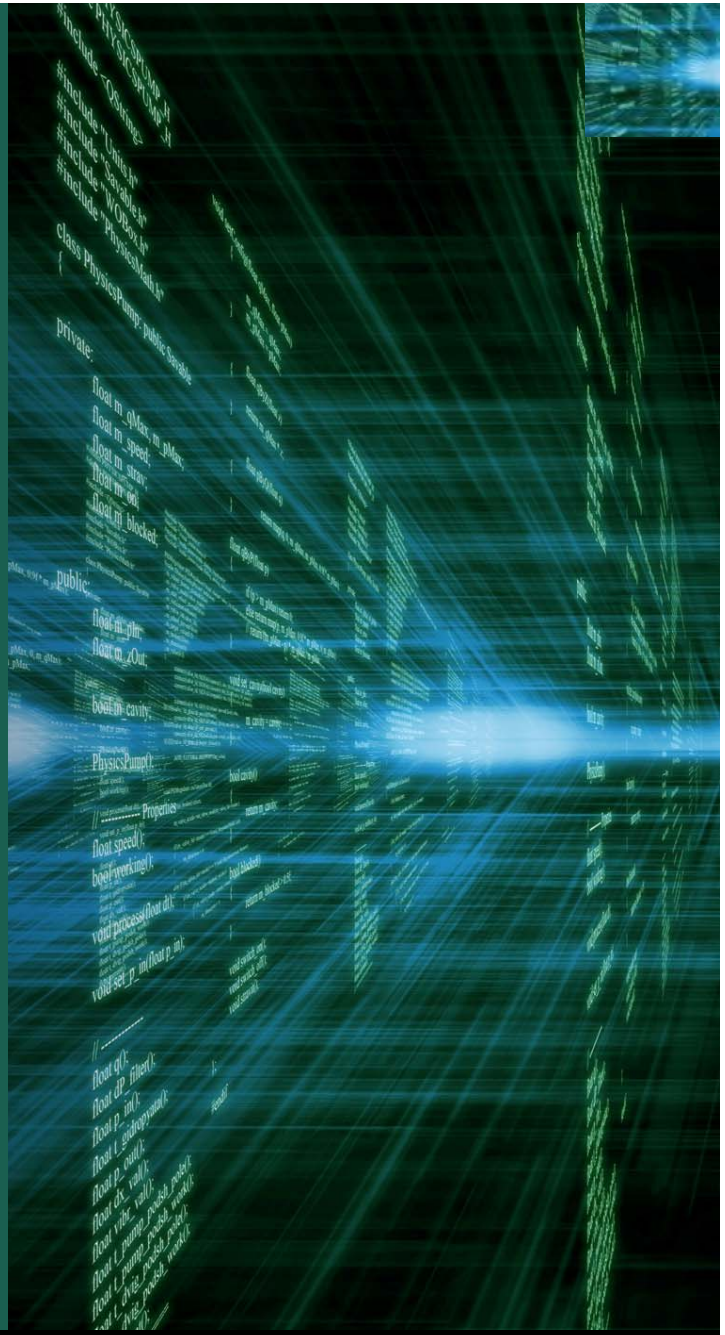
DM-0002873

Agenda

Set the context for the Enterprise IT System

Focus on five themes common to this and other Enterprise IT Systems

Context



System Context

Provide an electronic environment for the creation and management of records comprising complex documents or document sets (e.g., law suits, contracts, and patient records) and related information

- Contains both structured and unstructured data
- Intended for broad use throughout the organization (approximately 10,000 users)

System is being developed incrementally

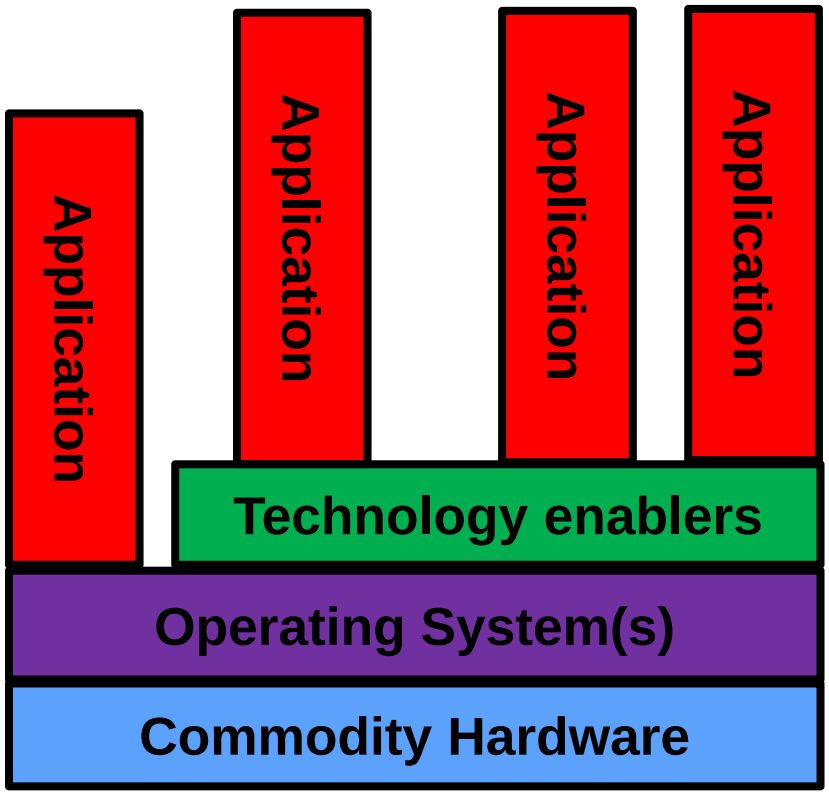
- Record creation is the first increment
- Even creation requires interactions between the system and other supporting systems

Architecturally, the system comprises infrastructure and a collection of applications to create and manage records

Infrastructure

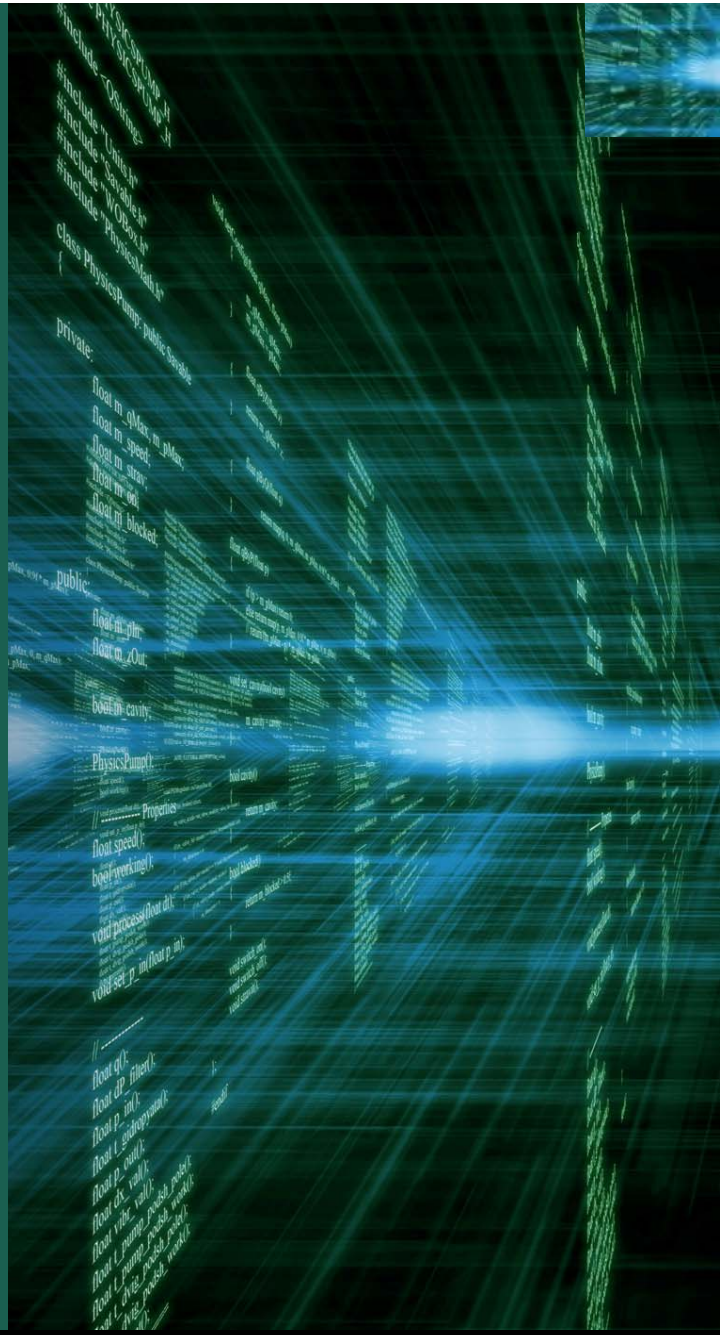
Expected to be built from COTS software components drawn from these technology enabler areas:

- business intelligence
- business logic/rules engine
- workflow and orchestration
- knowledge management
- common data standards/ontology
- rendering
- collaboration
- security services
- workload management
- data services
- discovery services
- messaging services
- electronic records management
- archiving
- presentation layer



- Operating System(s) and hardware might be provided in a cloud context
- COTS products preferred for technology enablers
- Preference for applications to use technology enablers

Common Themes



Themes Arising from Technology Investigation

Study of the different technology enablers led to these themes:

- Integrated COTS product suites and vendor lock-in
- Integrating application components in a Software-as-a-Service context
- Benefits and costs of using loose coupling and web services
- Trade-offs involved in achieving a “common look & feel” across the system
- Different levels of security services across the enterprise

COTS Product Suites

The technology enabler areas couldn't be considered independently of each other due to:

- Inherent overlap between enabler areas
- Product feature growth
- Corporate mergers

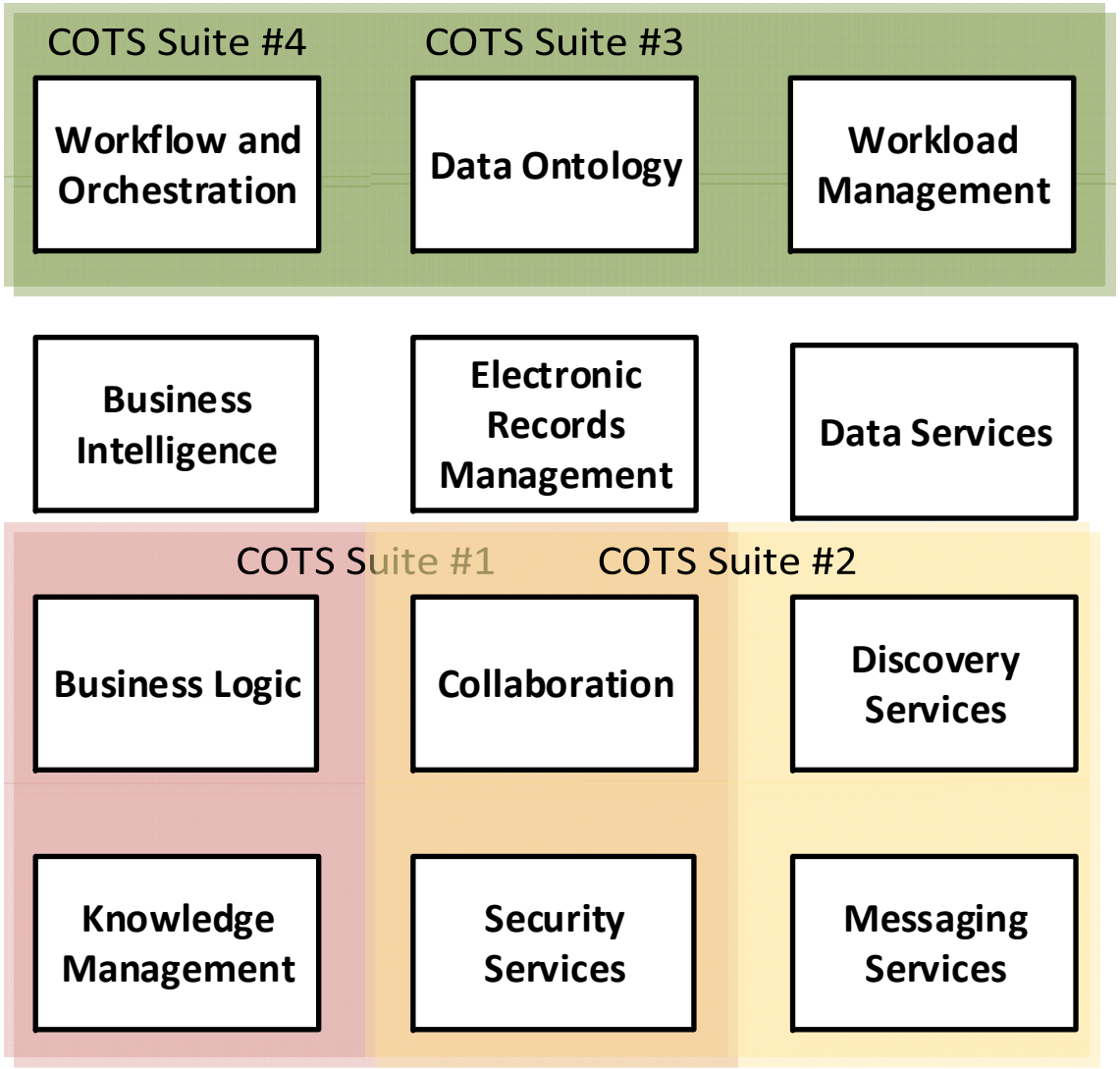
The consequence is the tendency is toward greater dependence on single vendors

- Even if products are initially independent of each other, a long-lived system will eventually depend on a few (maybe as few as one) products or vendors
- The benefit is that the cost of integrating products will tend to go down

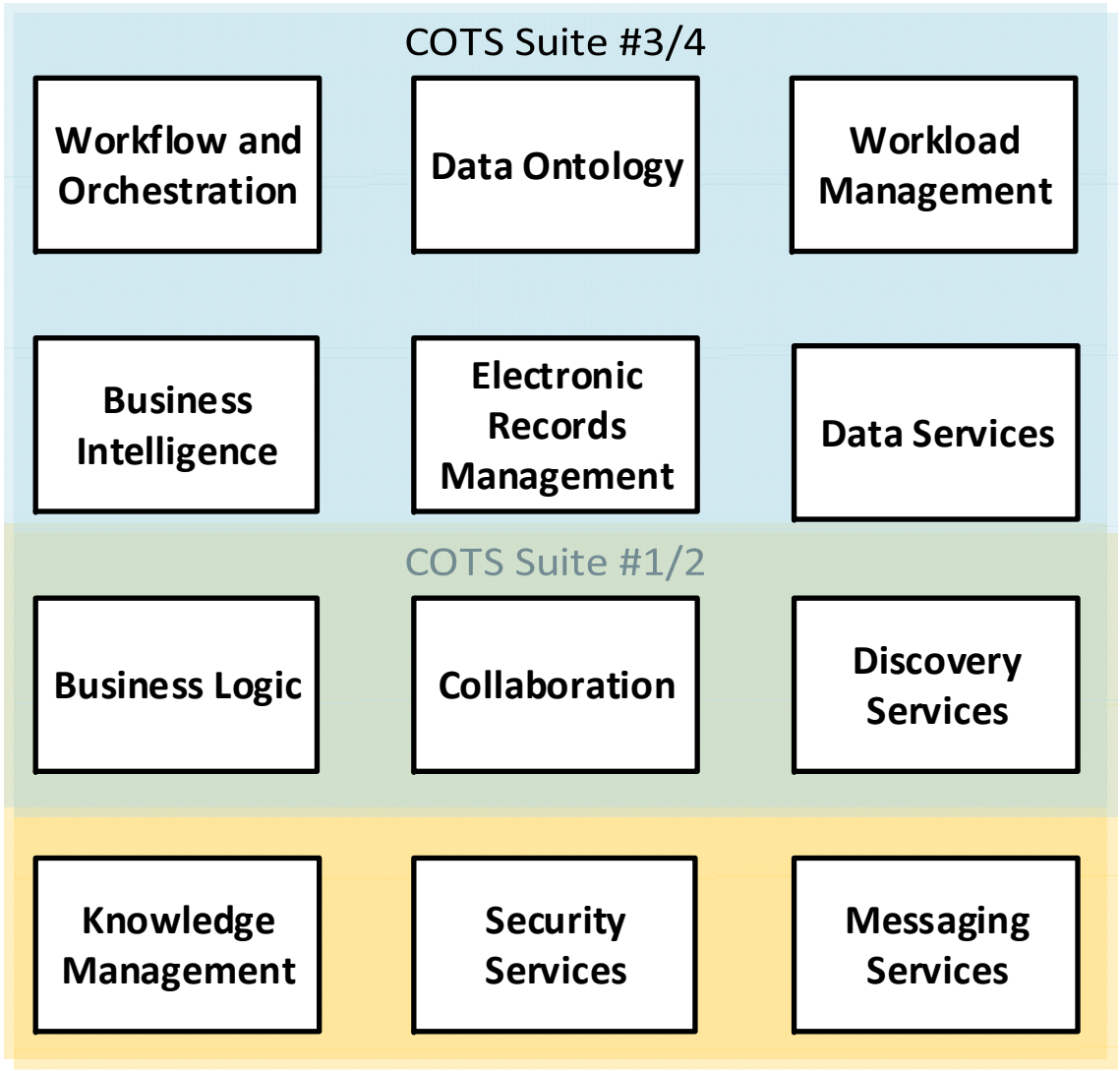
COTS Product Suites

Workflow and Orchestration	Data Ontology	Workload Management
Business Intelligence	Electronic Records Management	Data Services
Business Logic	Collaboration	Discovery Services
Knowledge Management	Security Services	Messaging Services

COTS Product Suites



COTS Product Suites



Software-as-a-Service (SaaS)

Strong preference for a cloud-based Infrastructure-as-a-Service

With common technology enablers can provide Platform-as-a-Service

Software-as-a-Service is the next logical step, but:

- Organizational policy is still changing
- Complexity when different components reside in different clouds
- Complexity of keeping applications synchronized when delivered by different providers
- Sensitivity of records
- Limited control over provider

Loose Coupling and Web Services

Highly desirable

- Makes it easier to replace components
- Achievable between the applications
- Ideally coupling is only through records (structured data can be a problem)
- Simplifies testing

Counter to the tendency towards product suites

Service orientation through use of web services still seems the best way to achieve loose coupling

Achieving a Common Look-and-Feel

Common look-and-feel is highly desirable from a user perspective but:

- If a style is chosen will either:
  - Limit the number of available products
  - Or require modification of the products to conform to the style
  - Or develop every component as custom
- If not chosen, then the first product defines the style
- Require everything be built to a custom style

Practically, particularly with an incremental acquisition for a long-lived system, a common look-and-feel is not possible

Security Services in an Enterprise Context

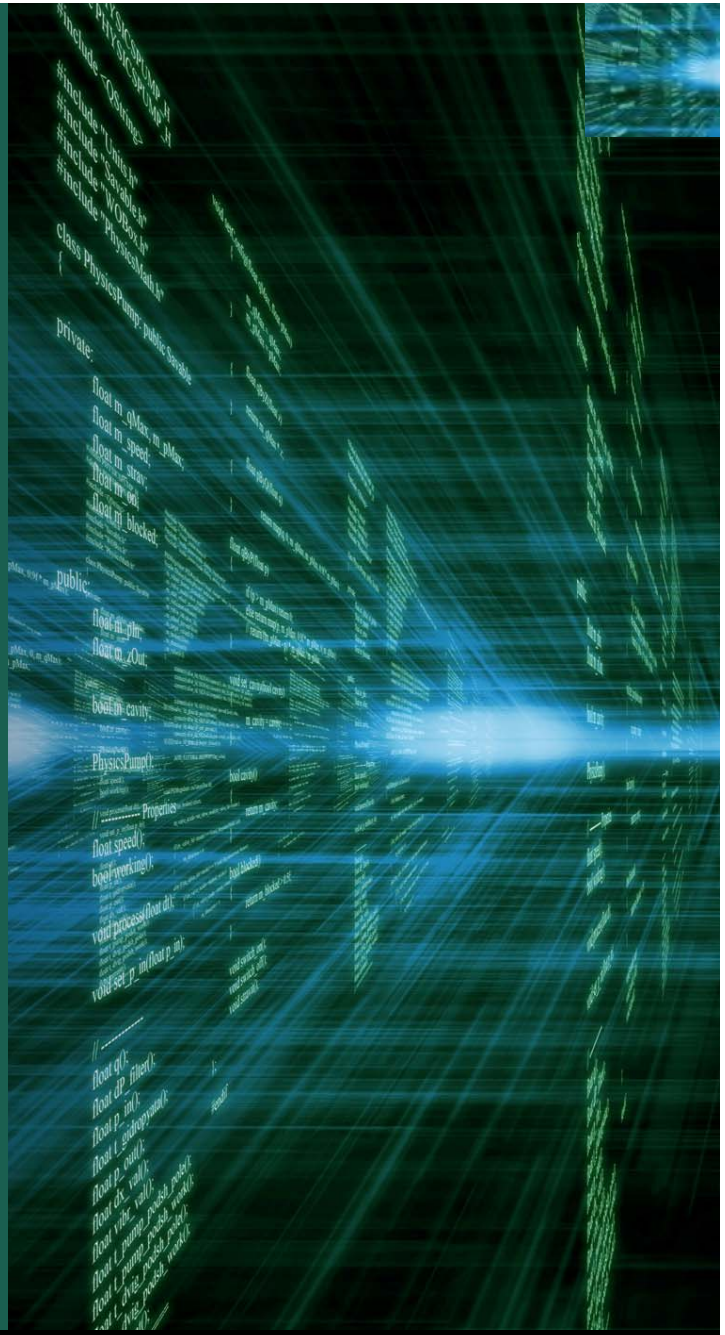
Identity and Access Management are still areas of development

- Name & password pairs not strong enough
- PKI technology is still being adopted

Choices for the system:

- Every application has own security capability
  - Least usable
- One application provides security for others
  - Can provide single sign on (SSO)
  - Have to find an application that can serve up security services
  - Have to adapt other applications to use it
- Provide separate security service
  - Neutral with respect to applications
  - Can provide SSO
  - More readily adapted when organization policy changes
  - More effort to develop
  - All applications need to be adapted to use the service

Conclusions



Conclusions

No theme is really new

- There are new aspects of them
- Their continued recurrence suggests there is still work to do

Smaller, more incremental, acquisitions will exacerbate issues

- May be unavoidable
- Predictions of the future are rarely accurate

Program Offices need to have realistic expectations

Contact Information

William E. Novak

Client Technical Solutions

Telephone: +1 412-268-5519

Email: wen@sei.cmu.edu

Patrick R.H. Place

Client Technical Solutions

Telephone: +1 412-268-7746

Email: prp@sei.cmu.edu

U.S. Mail

Software Engineering Institute
Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612 USA

Web

www.sei.cmu.edu
www.sei.cmu.edu/contact.cfm

Customer Relations

Email: info@sei.cmu.edu
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257