Using Standards to Achieve Commonality

Judy Kerner

The Aerospace Corporation

Ground System Architectures Workshop

March 4, 1999

Outline

- Why commonality?
- Anticipated vs. unanticipated interfaces
- Kinds of standards used
- How effective are they?
- Future directions

Why Commonality?

- High cost of "stovepipe" systems
  - Can't take advantage of large-scale reuse
  - Even component-level reuse may be challenging
- Resources are decreasing
  - Takes too long to build complete systems
  - Can't afford costly operator retraining
  - Requirements change dynamically
- Interoperability and portability are increasingly important requirements of system development

Anticipated and Unanticipated Interfaces

- If you know System A will interface only with System B, you can jointly create one ICD
- But if you don't know what System A will be deployed with, or who will operate it, then
  - If you follow a common approach, you can at least plan to interface with others following the common approach
  - If there is a standard to follow, you can reduce the n-way problem to many one-interface mappings

Kinds of Standards

- Specifications may be standardized
  - POSIX, SQL, Motif are standards
  - DoD Joint Technical Architecture (JTA) identifies standard specifications for many service areas
- Standard products assure some amount of commonality
  - Defense Information Infrastructure (DII) Common Operating Environment (COE) provides middleware functionality
    - Mission applications can be ported
    - Some interoperability with other COE-based systems assured

Are Standards Effective?

- Some standards are in widespread use
  - TCP/IP, Java, HTML are critical to the "Internet revolution"
  - POSIX and SQL are used by many commercial systems
  - USMTF, NITF support military information exchange
- Standards create challenges for the adopter
  - JTA standards don't always support needed functions
  - COE doesn't provide real-time performance and raises many security concerns
  - Any common product becomes an architecture driver

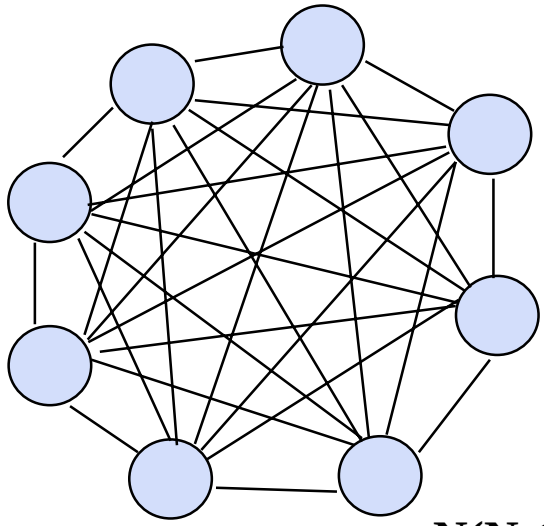
Future Directions

- There is widespread agreement that we can't afford stovepipe systems any more, but
  - Proposed solutions are often in conflict
  - Tailoring standards to meet requirements may reduce or eliminate the commonality benefit
- Commercial industry (and increasingly, DoD) is focusing on reducing cycle time
  - Standard architectures and components facilitate this
- Product lines and "system-of-systems" approaches are gaining increasing acceptance
  - Many domain-specific reference architectures exist

Backup Charts

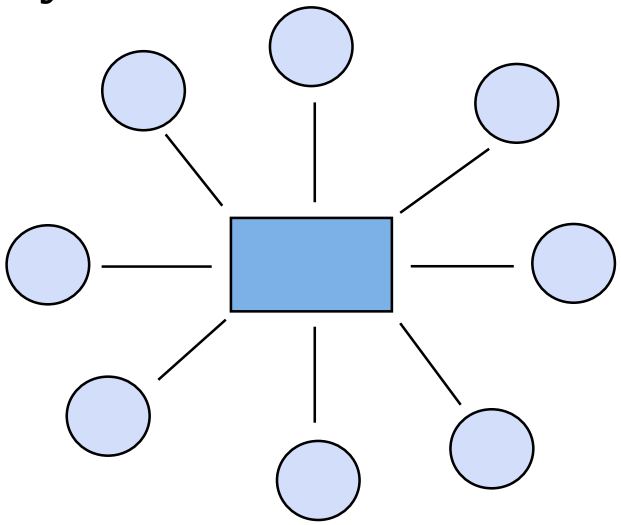
Achieving Interoperability

by Interface Ctl Doc (ICD)



- # interfaces needed = $\frac{N(N-1)}{2} = N^2$
- Evolutionary path prohibitive
- Never gets cheaper
  - next system needs N + 1 interfaces
  - next generation needs N^2 interfaces

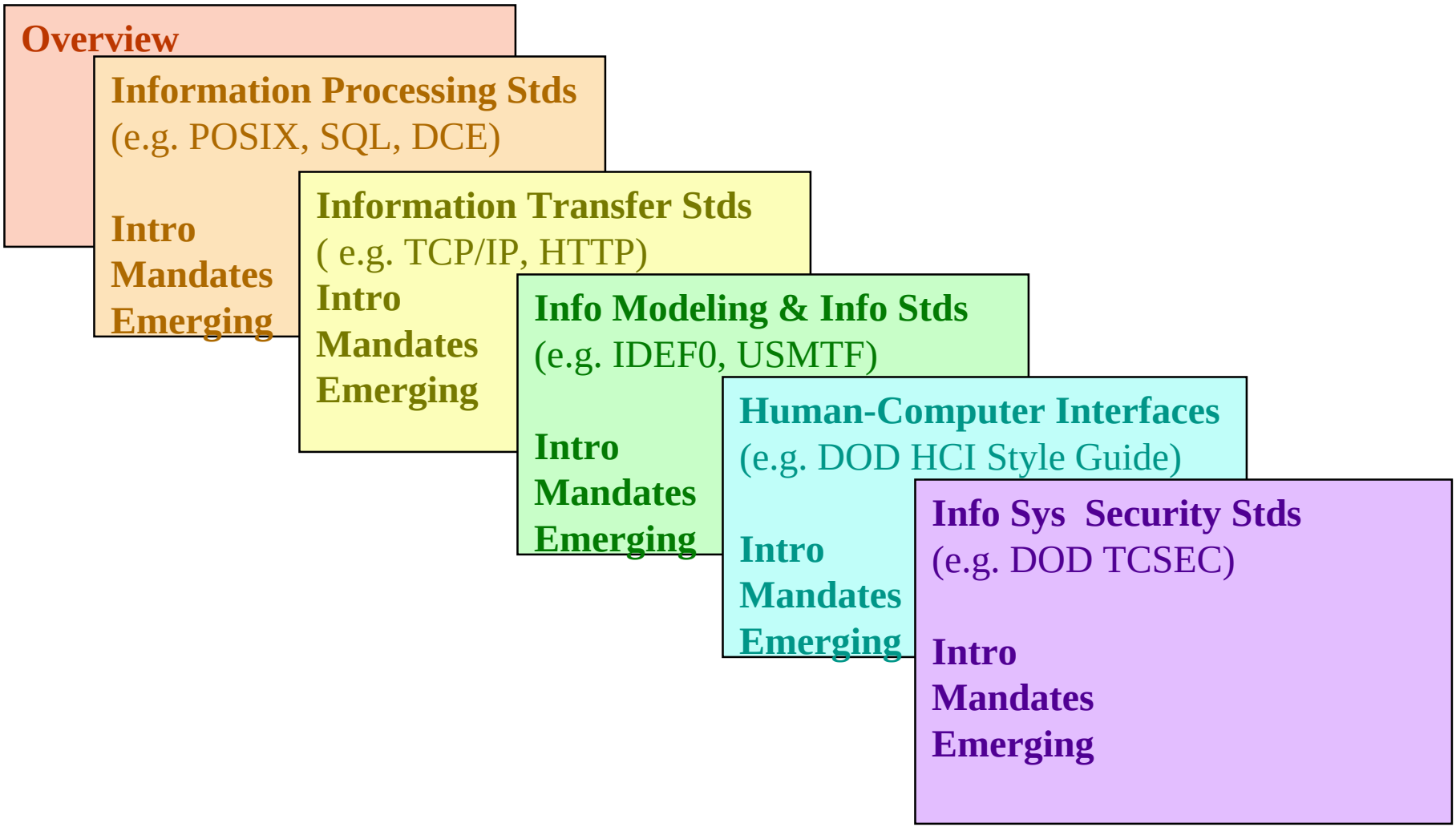
by interface standards



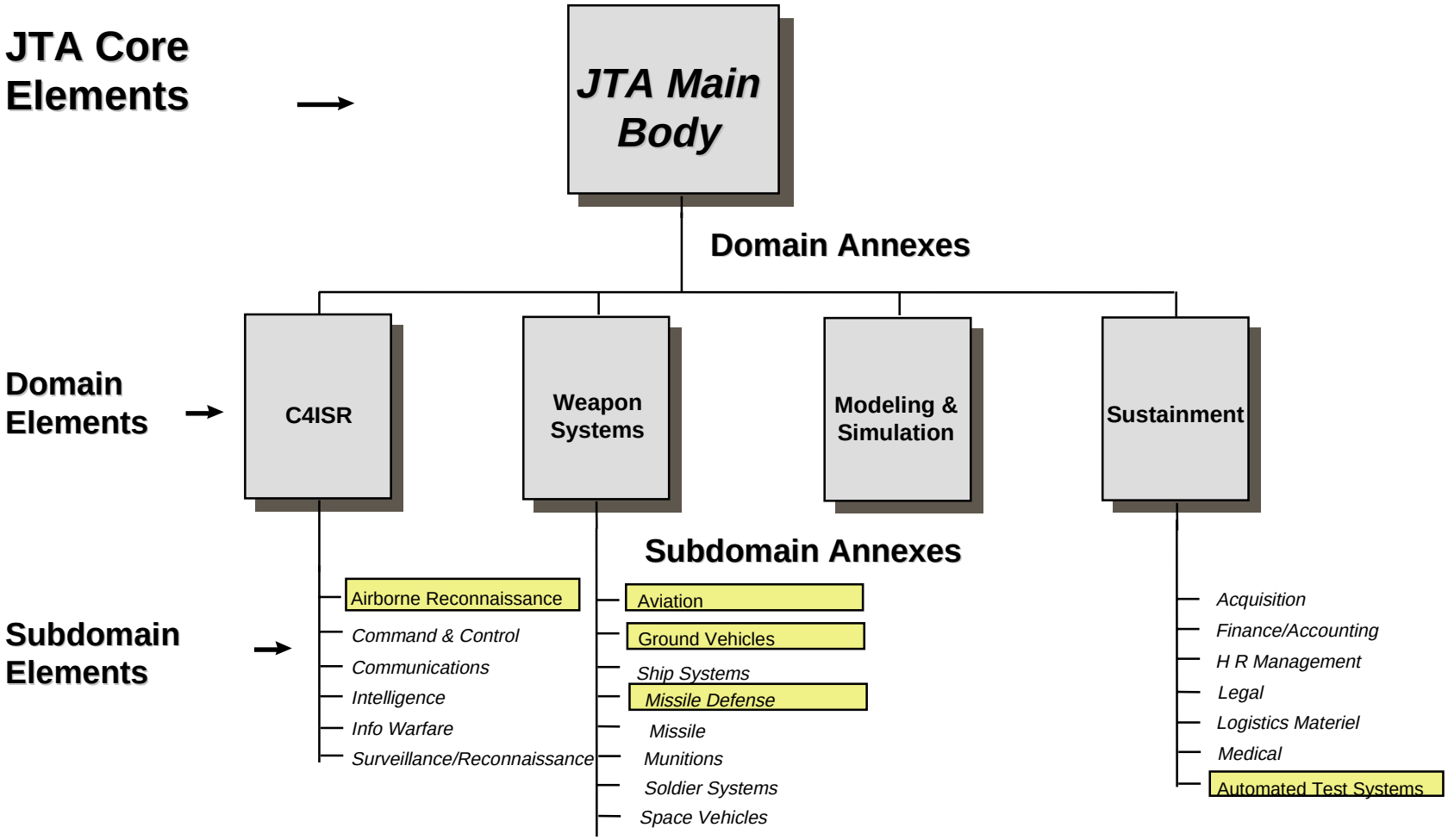
- # interfaces needed = N + 1
- Evolutionary path identified
  - next system needs 1 interface std
  - next generation needs 1 interface std

Can’t know in advance all the systems that will have to interoperate

JTA Element Structure

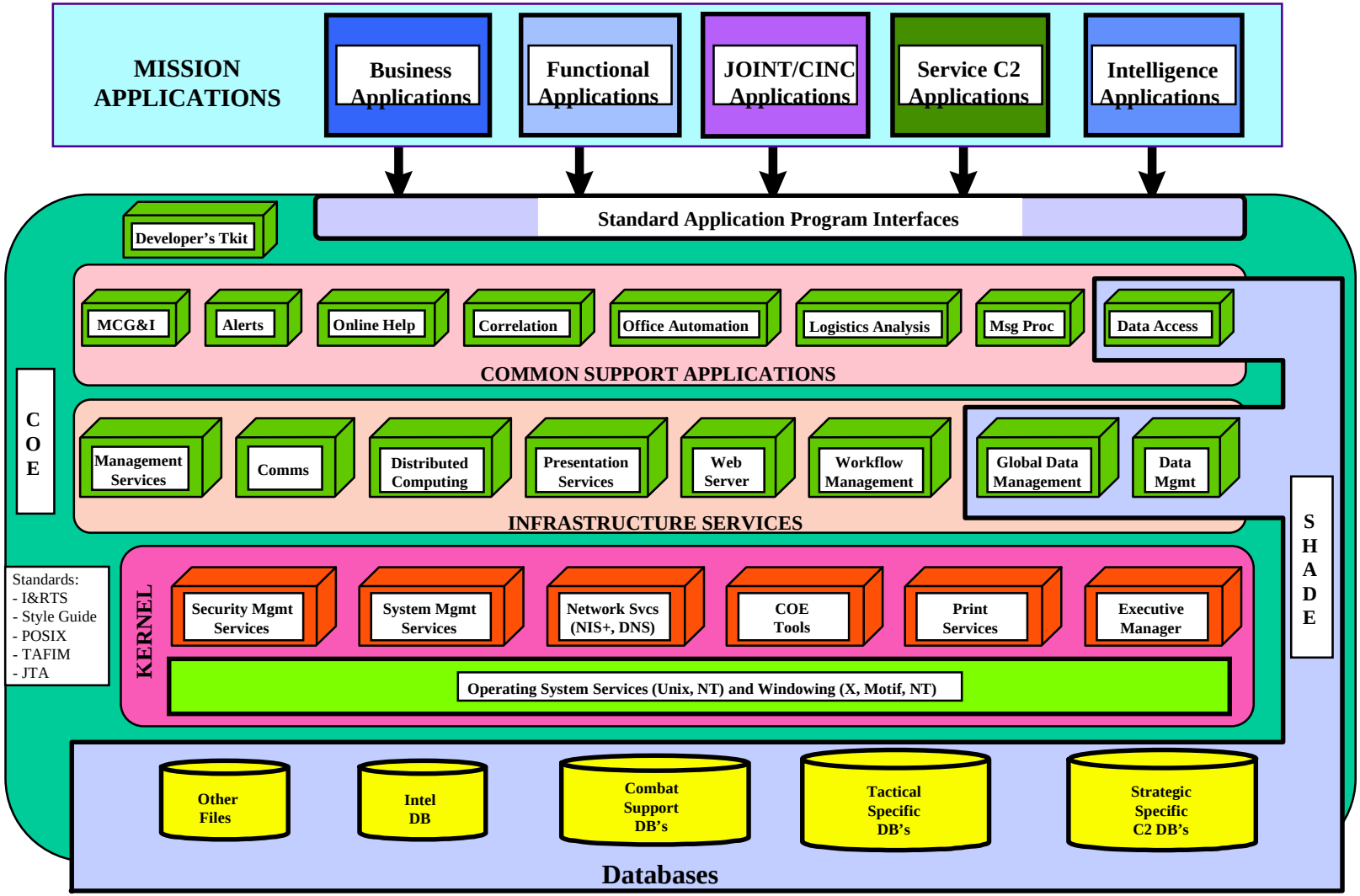


JTA Version 2.0 Organization



(From JTA V2.0)

DII COE



(From DII COE I&RTS V3.0)