M&S Composability

SCS 2003 – Sim Summit

Phil Zimmerman, DMSO (USA)

Assoc Dir, Composable Mission Space Environments

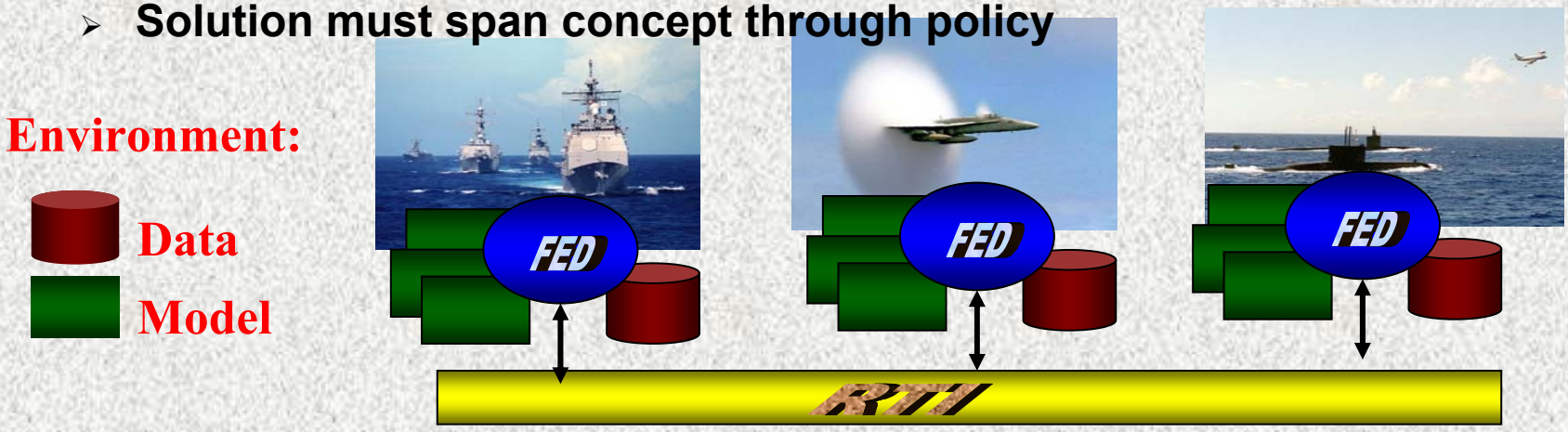
(slides shamelessly borrowed from SKN)

State of the Practice

- Need to reduce the number of "geek weeks" required to build a federation and a scenario
  - Takes too long to develop and implement large scale distributed modeling and simulation (e.g., $250M and 18 months) and payoff in terms of execution time is small, and not "persistent"
- It takes a battalion to train a battalion
  - Human controllers spend time correcting errors and adjudicating actions that should be correctly executed in appropriately developed simulation components.
  - Simulation is pervasive, but human cost is too high
- Singular point solutions are pervasive
  - Multi-resolution simulation is a brute force process that attempts to move context as well as bits
  - Simulations access live systems in multiple, non-standard ways
- HLA is a necessary but not sufficient enabler for interoperability and reuse
  - MC02 – "One of the difficulties of creating federations is that interactions between federates must be grounded in common or at least compatible models in all federates"
  - (OBTW, DMSO has not given up on HLA)

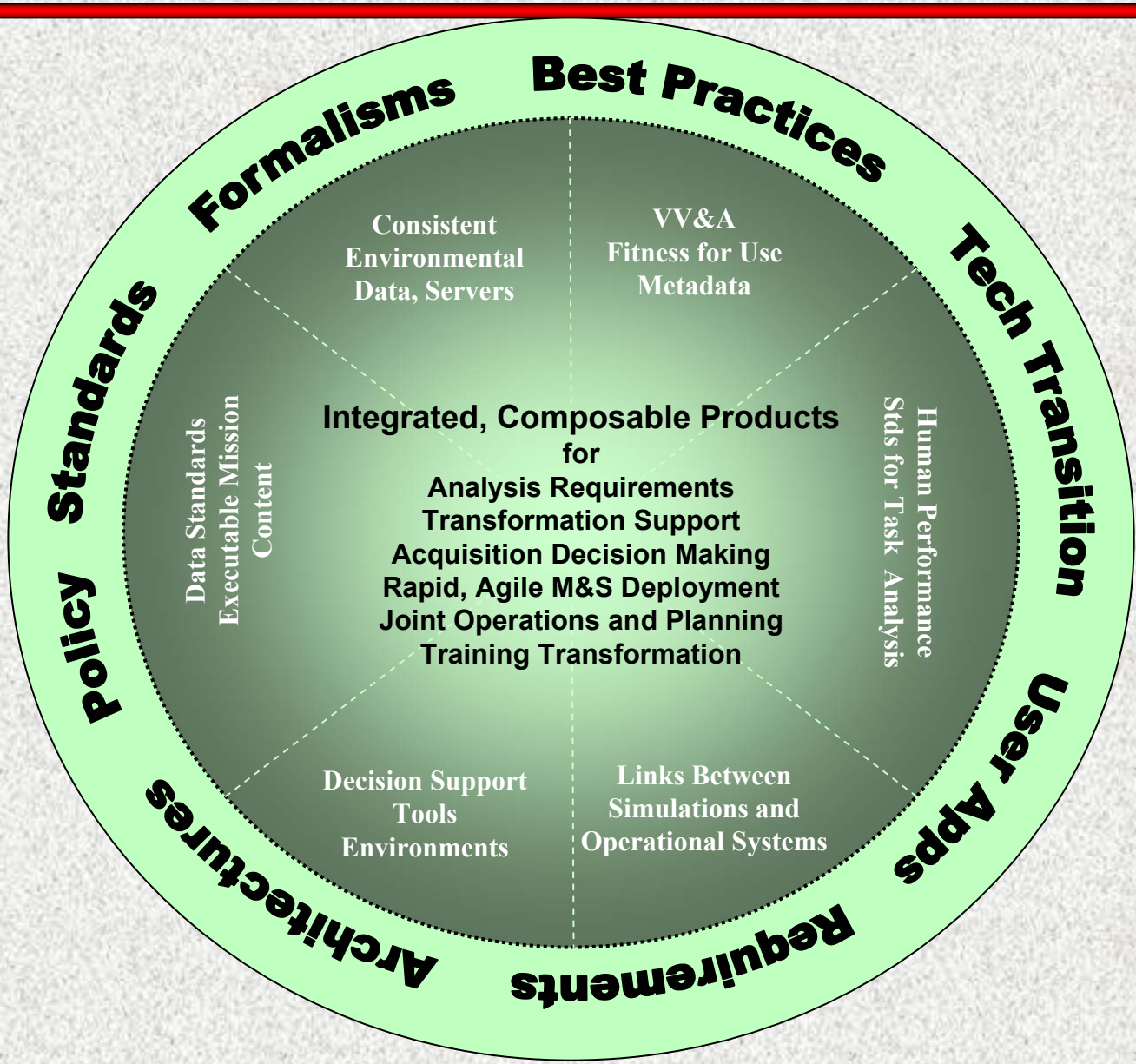
Challenge: Composability

- **Independent Development / Shared Dependencies**
  - Mix of systems results in spanning multiple orders of magnitude
  - Cannot just plug in valid pieces and expect good answers
- **Authoritative Data Is Not Enough**
  - Common certified data, validated models -- insufficient
  - Validity required across all systems – not just for each system separately
- **Solution Implies More than Current Capability**
  - Well-define, composable modules
  - Design specs for such modules
  - Metadata, tools, methods for using composable modules
  - Solution must span concept through policy



Composability as an Attractor

- **Integration and Interdependence are Key**
  - Diverse parts must be composable
  - Answer is at least partially dependent on standards, basics
- **Components are built from elements of related, but different technical areas**
  - Metadata is common
  - VV&A designed in, not measured after the fact
  - Common data must address all pieces
- **Technical barriers and subject areas blur in the development of composable pieces**
- **Requirements drive integration**
- **Must look to the non-M&S domains**



Component Attributes

- **Usability and automation**
  1. May be used by other software elements (clients).
  2. May be used by clients without the intervention of component developers.
  3. Is easily composable with other components.
     - Well documented interfaces, data requirements, etc.
  4. Can be integrated into shared, collaborative environment.
- **Documentation: metadata**
  5. Includes a specification of all dependencies (hardware and software platform, versions, other components).
  6. Includes a precise specification of the functionalities it offers.
  7. Is usable on the sole basis of that specification.
- **What is a component?**
  8. Components are NOT necessarily objects in the OO sense.
  9. Not just software (hardware and data too)
  10. Open source desirable but not required.
- **Composability is not just an architectural concept**
  - It involves all the concepts and tools we currently use, and more
  - It encompasses data (real-time delivery, mission content, descriptive)
  - It relies more than ever on conceptual models and validation
  - Composability is a central focus for simulation development
  - MUST BE FORMALLY EXPRESSED!!!!

Sample Research: Domains of Validity

- What if we could mathematically define domains over which simulation objects could be composable?
- Would domains equate to "application domains"?
  - Analysis, Acquisition, Training are overlapping sets – poor definition for our purpose
  - Resolution, granularity, computational context may be more appropriate parameters for defining sets
  - Model or interaction as the basis – *(DSTO- Grisogono)*
- If we can define the domains over which composition is valid, can we bridge between domains using generalizable means?
- Can we define characteristics of modules and workspaces that aid the user in creating consistent wholes from composable pieces?
  - The processes for composing simulations should include
    - Software automation/assistance
    - Error detection and correction
    - Computer-aided repository searches
    - Risk mitigation assistance

Barrier: Changing Our Habits

- Focus of reuse is traditionally at the implementation level vs the design level
  - If we can define domains over which composability is valid, that definition must be part of the conceptual model
  - The definition must lead design, not follow it
- Appropriate components hard to find and evaluate
  - Is that because we have failed to define our "sets" adequately?
  - How do you define suitability of components?
- Documentation often inadequate, and unparseable
  - Have we determined what key factors must be documented?
- Economic balance between legacy and new designs
  - Our success will be limited by the need to use "unmodernized" legacy models.
- Subtle differences in interfaces can lead to insidious errors

-- Heineman, George T. and W.T. Councill, Component-Based Software Engineering: Putting the Pieces Together, Addison-Wesley, Boston, 2001.

Look Who's Talking….

M. D. Petty. "Two Aspects of Composability: Lexicon and Theory," DMSO Composable M&S Workshop July 9-10, 2002. (see also Spring SIW)

S. Kasputis, and H.C. Ng. "Composable Simulations," *Proceedings of the 2000 Winter Simulation Conference,* pp. 1577-1584, Dec. 2000

J. Aronson and D. Wade. "Benefits and Pitfalls of Composable Simulation," *Proceedings of 2000 Spring Simulation Interoperability Workshop.*

P. Davis, P. A. Fishwick, M. Overstreet, and D.Pegden. "Model Composability as a Research Investment: Responses to the Featured Paper," *Proceedings of the 2000 Winter Simulation Conference.*

B. Ziegler. "Support for Hierarchical Modular Component-based Model Construction in DEVS/HLA," *Proceedings of 1999 Spring Simulation Interoperability Workshop, 99S-SIW-066.*

D. Pratt and C. Ragusa. "Composability As an Architecture Driver," *Proceedings of the 1999 I/ITSEC Conference.*

S. Harkrider, and D. Lunceford. "Modeling and Simulation Composability," *Proceedings of the 1999 I/ITSEC Conference.*

E. Page and J. M.Opper. "Observations on the Complexity of Composable Simulation," *Proceedings of the 1999 Winter Simulation Conference* pp. 553-560, 1999.

M. Biddle and C. Perry. "An Architecture for Composable Interoperability," *Proceedings of the 1999 Simulation Interoperability Workshop.*

E. H. Page and J. Opper. "Theory and Practice in User-composable Simulation Systems," DARPA Advanced Simulation Technology Thrust (ASTT).

J. Aronson. "Model-based Simulation Composability," *Proceedings of 1998 Fall Simulation Interoperability Workshop,* 98SF-SIW-055.

R. Garlan, R. Allen, and J. Ockerbloom. "Architectural Mismatch or Why It's Hard to Build Systems Out of Existing Parts," *Proceedings of the Seventeenth International Conference on Software Engineering. April 1995.*

There are other: Doubters, Enthusiasts, Critics