



U.S. Army Research, Development and Engineering Command



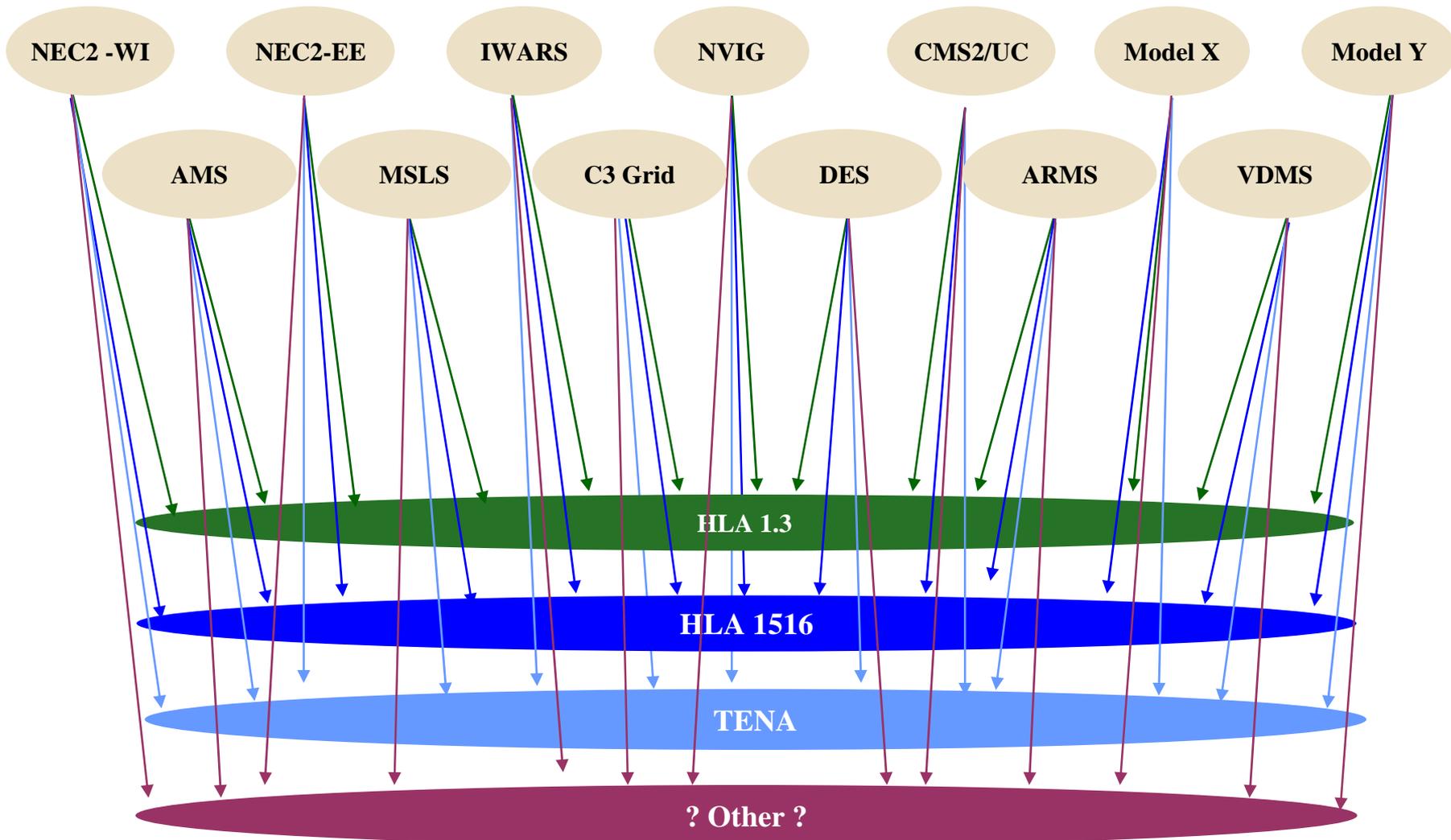
***TECHNOLOGY DRIVEN. WARFIGHTER FOCUSED.***

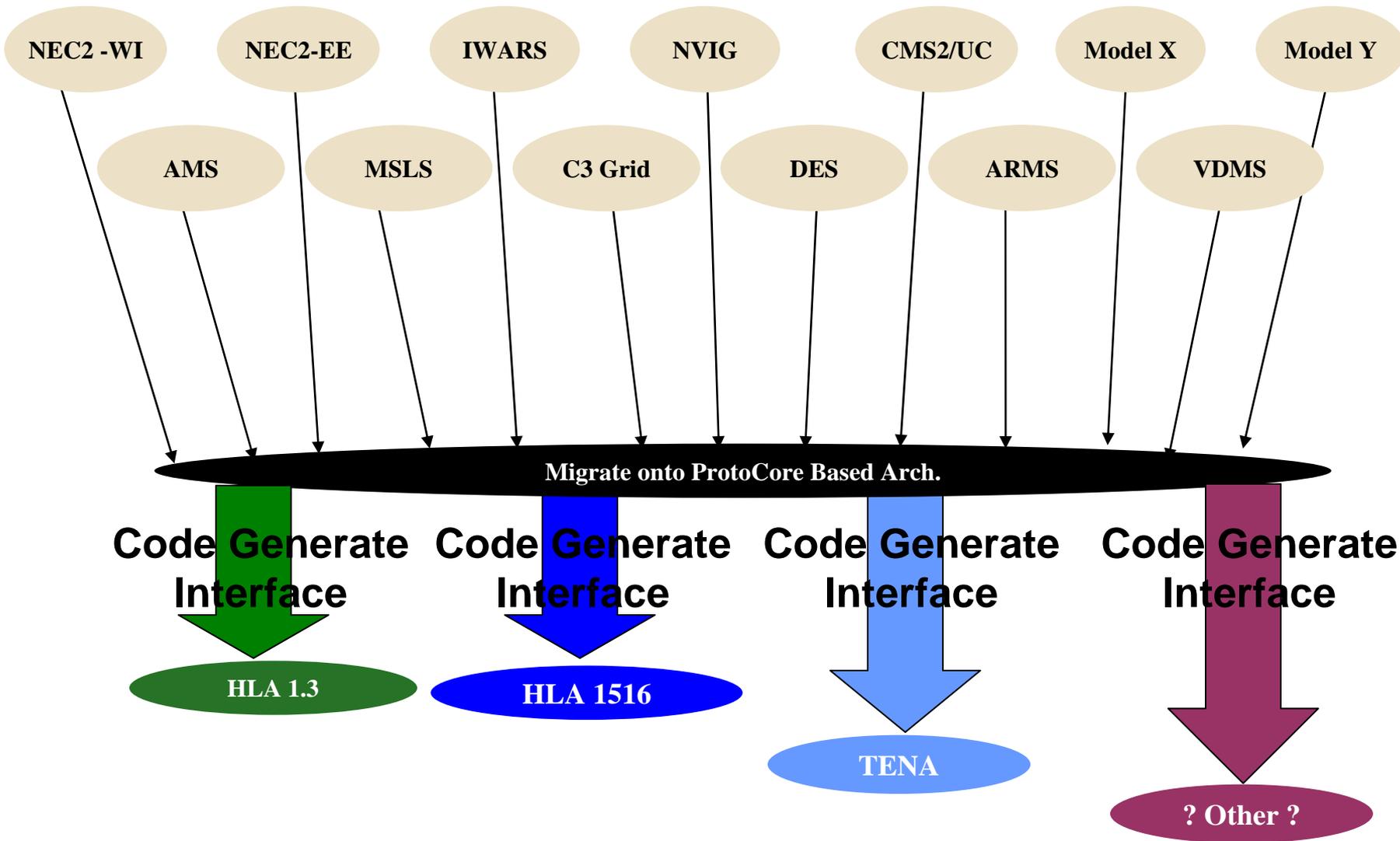
ProtoCore

DoD M&S Conference

10 March 2008

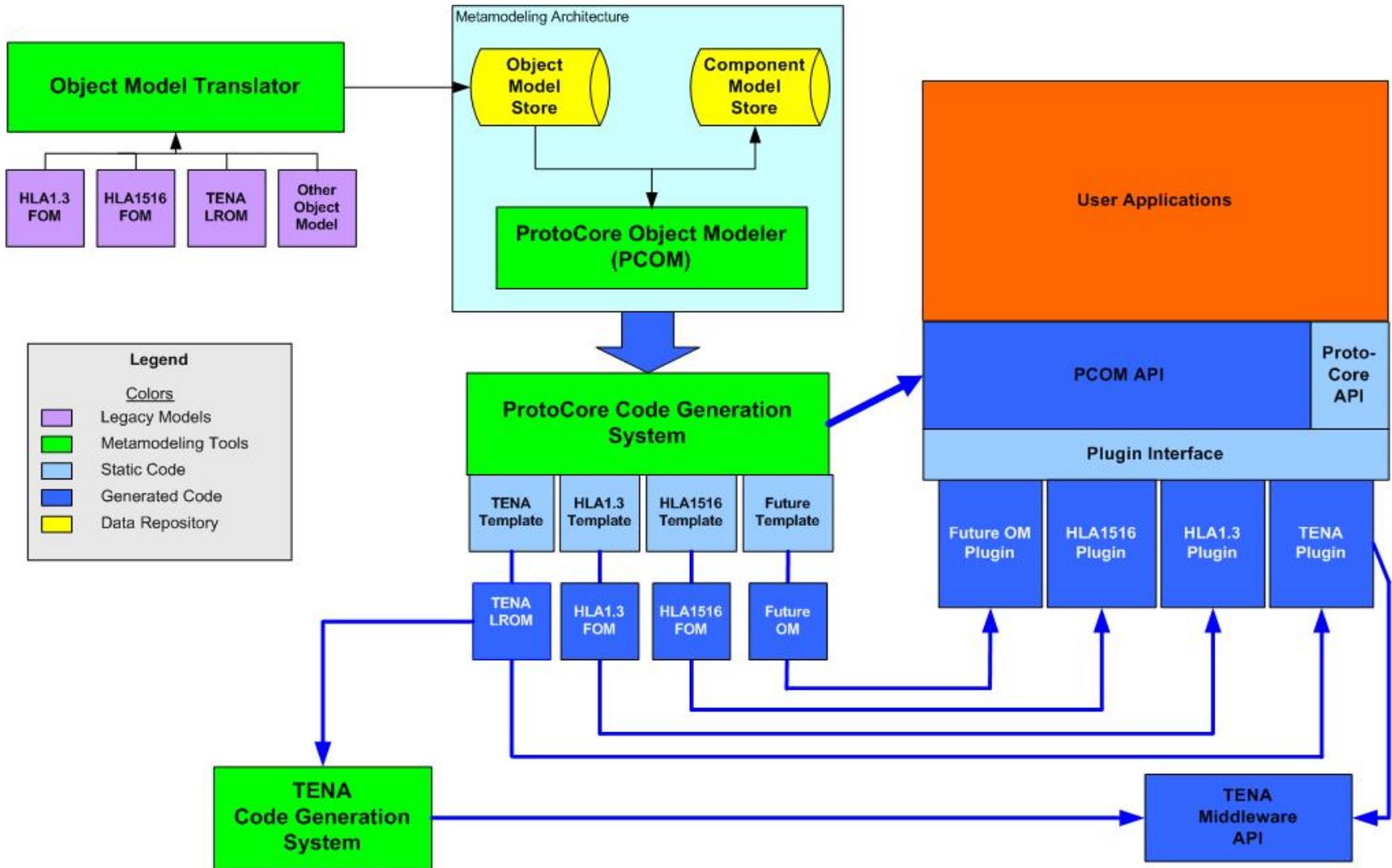
- Legacy middleware architectures, used in many simulation environments, do not make use of modern programming practices and can be cumbersome and error prone to use. e.g. HLA
- More and more middleware architectures are being used within the Army and DoD Modeling and Simulation for various purposes.
  - Existing infrastructures continue to be used for legacy simulations which have invested a lot of resources developing and maturing that environment. Switching is difficult and expensive.
  - Middleware to be used for given exercises is unknown and often open to change.
  - The application developer potentially needs to be an expert user for all existing and emerging architectures: e.g. HLA 1.3, HLA 1516, TENA, DIS, OOS
- Application developers are spending more and more time porting applications between various middleware architectures to enable Live-Virtual-Constructive (L-V-C) experimentation.







- Provide modern Object Oriented and type safe Application Programming Interface (API) to distributed simulation services
  - Provide automatic handling of menial tasks, such as data marshaling, required for some infrastructures.
  - Type safe API allows more errors to be caught at compile time rather than runtime.
- Provide mechanism to connect API to various network protocols in near-term and minimize work of moving applications between them
  - Code-Generate connectivity to protocols from a common Object Model
  - Plugin architecture used to allow single application binary to run over various protocols without modification.
- Leverage existing software investments and minimize impact
- Provide a forward path for legacy object models and component models into the future



- Application Portability
  - Able to be used with multiple different transport protocols
  - Plugins only a single application binary simplifying deployment
- Preservation of Object Model
  - Existing object models can be used with and migrated to newer transport protocols
- Existing tools and applications can be migrated to new transport protocols
  - Preserve capability of existing models
  - Preserve test cases and capability



- **Code Generator**
  - Allows users to compile libraries for customized FOMs
  - Allows users to create custom templates for generating specific applications from Object Model
- **Functionality**
  - Simulation Management
  - Object Management
  - Interaction Management
  - Synchronization Points
  - DDM
  - Multiple Encoding/Decoding Schemes
- **Current Plug-ins**
  - HLA 1.3 Plug-in
  - HLA 1516 Beta Plug-in (Dynamic Link Compatible (DLC) API)
  - Prototype of TENA MW Plug-in
- **Java Bindings**
  - Supports JDK (Java Development Kit) 1.5+

- **Functionality**
  - Improve dynamic DDM support
  - Remote Method Invocation
  - Time Management
  - Ownership Management
  - Object Model Agility
  - Object Model Translator
- **Modeling Architecture**
  - Enhance ProtoCore Meta-Model to reflect additional semantic information captured within MATREX SDD Information Architecture
  - Provide tools to translate existing object models
- **Plugins**
  - Complete the HLA 1516 plugin (completed – in verification)
  - Complete the TENA plugin (completed – in final testing)
  - SoSCOE and DIS plugin in design and planning stage

- Current

- RDECOM (RDEC's)

- Aviation and Missile Research, Development and Engineering Center (AMRDEC)
    - Army Research Laboratory (ARL)
    - Natick Soldier Research, Development and Engineering Center (NSRDEC)
    - Tank and Automotive Research, Development and Engineering Center (TARDEC)

- Naval Research Laboratory (NRL)

- OASES

- FCS/LSI

- Mule
    - ANS

- ATEC

- DCARS
    - OT-TES
    - STORM
    - IMASE
    - ExCIS FSA

- Future

- TRADOC

- Battle Laboratory Collaborative Simulation Environment (BLCSE)
    - Mounted Maneuver Battlespace Lab (MMBL)

- ATEC

- JANUS

- Additional RDEC (ARMS, MSLS)

- Armament Research, Development and Engineering Center (ARDEC)



# Points of Contact



MATREX IDE Website: <https://www.matrex.rdecom.army.mil>



# BACKUP

- API distributed in binaries within the MATREX Tools release
- User and programming documentation exists in HTML format in tools release. Documents are generated from comments in code to help ensure it is kept current
- Support reflector and bug database for user support