

STEP Extended Architecture

AIA – ASD Interop. Coordination confcal
25th of November 2019

Jean Brange, Boost Conseil
Tom Bluhm, Boeing



? is STEP a complex sytem that we need to extend and maintain for decades ?

© PDES, Inc. 2018

One man
is directly in charge
of all stages from
requirements
to disposal



First STEP dev in 1985



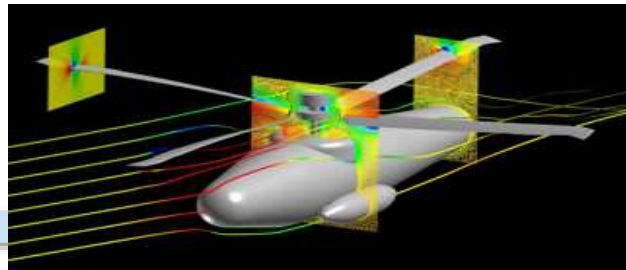
Work specialisation
Multiple human beings
Collaboration needs

*multiple AP with
integration requirements in 1995*



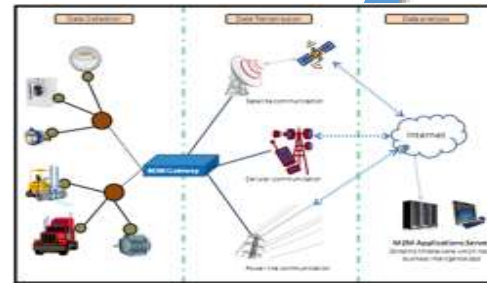
The man & the machine
interfaces

*Stepmod publishing
infrastructure for STEP*



Complex systems
Computing, MBE

*Extended Architecture
for model based
standards dev and publication*



Machine to machine
IoT
*integration
of standards families and
infrastructure
for stronger enterprise
integration*



~ 10 AP

350 Modules

> 65000 files

**1 000 000
pages**

STEP *iso*
10303

We need a strong and flexible framework to ensure the sustainability of our standards

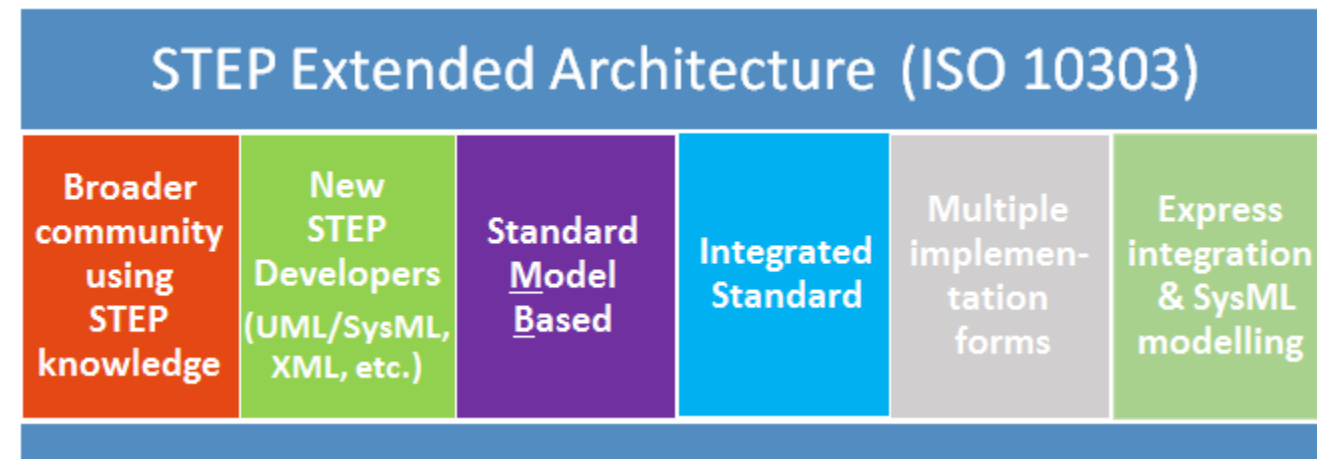
STEP Extended Architecture: upward compatibility with STEP legacy + support of new industry requirements

• Context

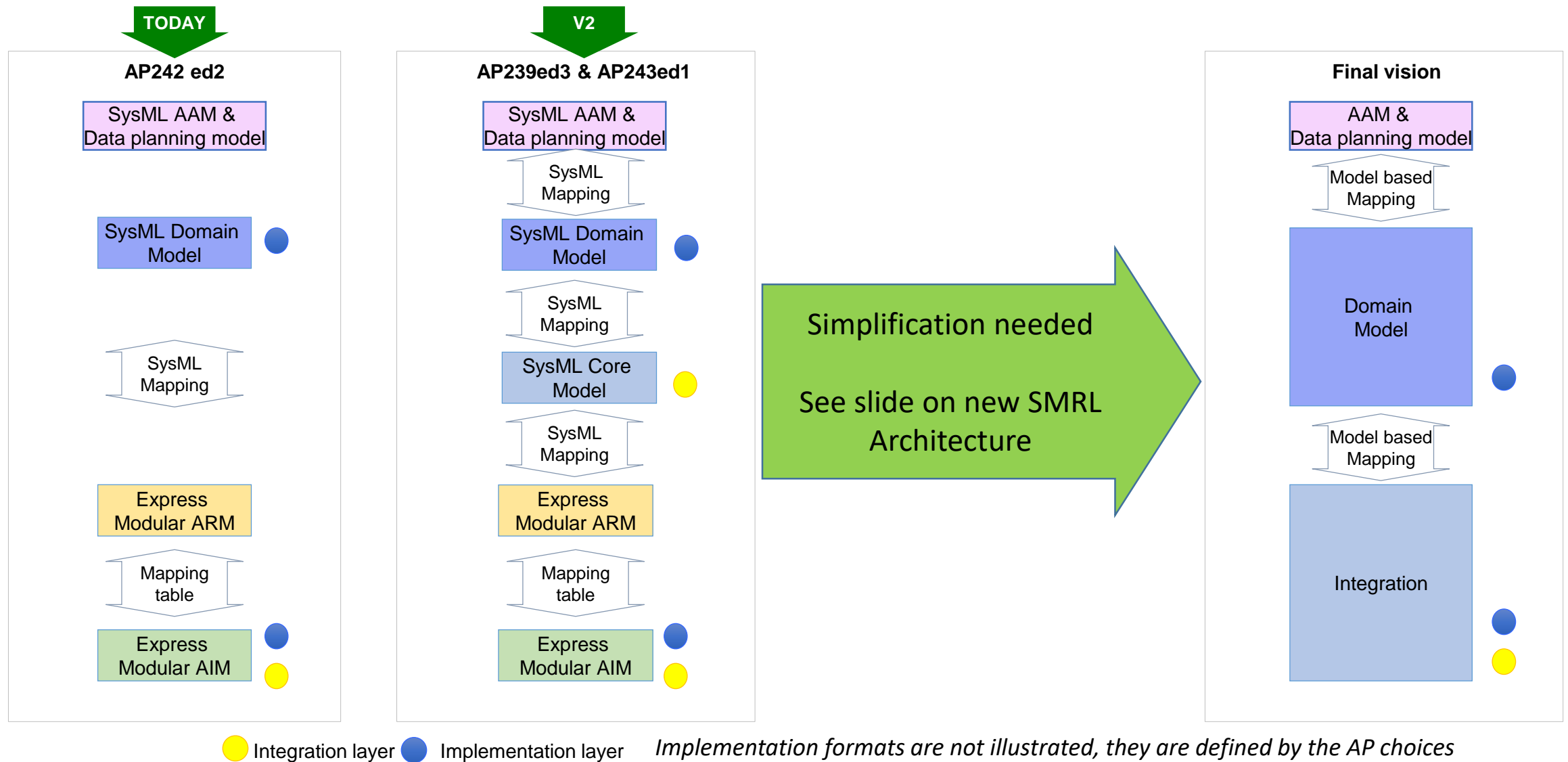
- Product Life Cycle → Systems Life Cycle : Increasing use of SysML !
- Systems Engineering → MBSE
- Increasing importance of electronics and electrical simulation, as part of mechatronics : SysML extension to support Physical Interaction and Signal Flow Simulation specification

• The answer:

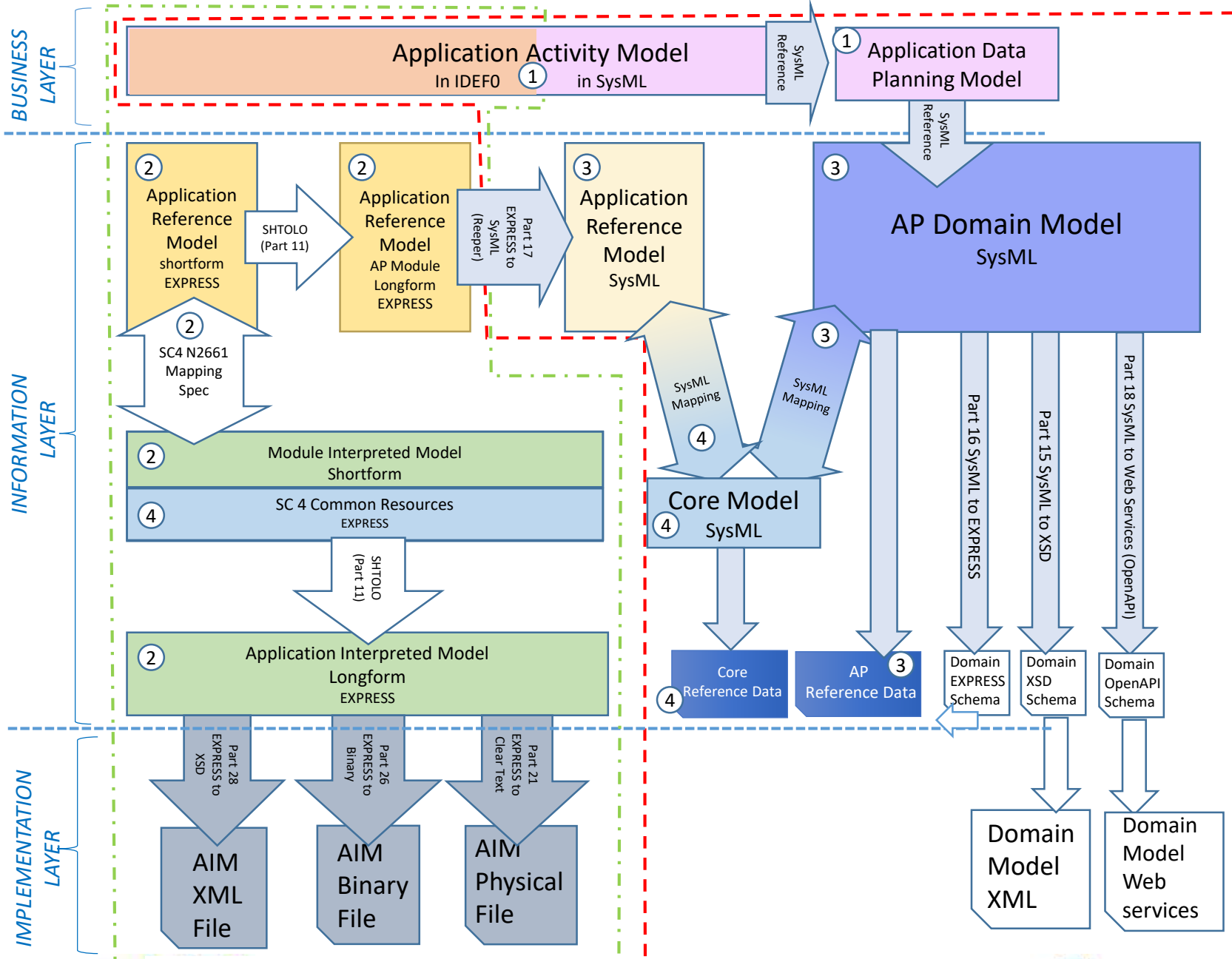
- To **extend the modelling framework of ISO 10303 in order to support;**
 - new industrial needs and
 - evolutions of Information Technologies



The development of ISO AP242 ed2, AP239 ed3 PLCS and AP243 MoSSEC relies on the implementation of the STEP Extended Architecture, in a phased approach



We are in a transition phase with a necessary overhead for managing the legacy and future state together. We need to converge to the target state as soon as possible



V2

- Where Key Models are Standardized:
- ① Application Protocol 10303-200s
 - ② Application Module 10303-400s
 - ③ Application Domain Model 10303-4400s
 - ④ Resource parts 10303-40-60s, -100s, -500s, - 4000

We shall reduce the complexity of our current overhead



Scope of initial
STEP Architecture



Scope of Extensions to
STEP Architecture V2

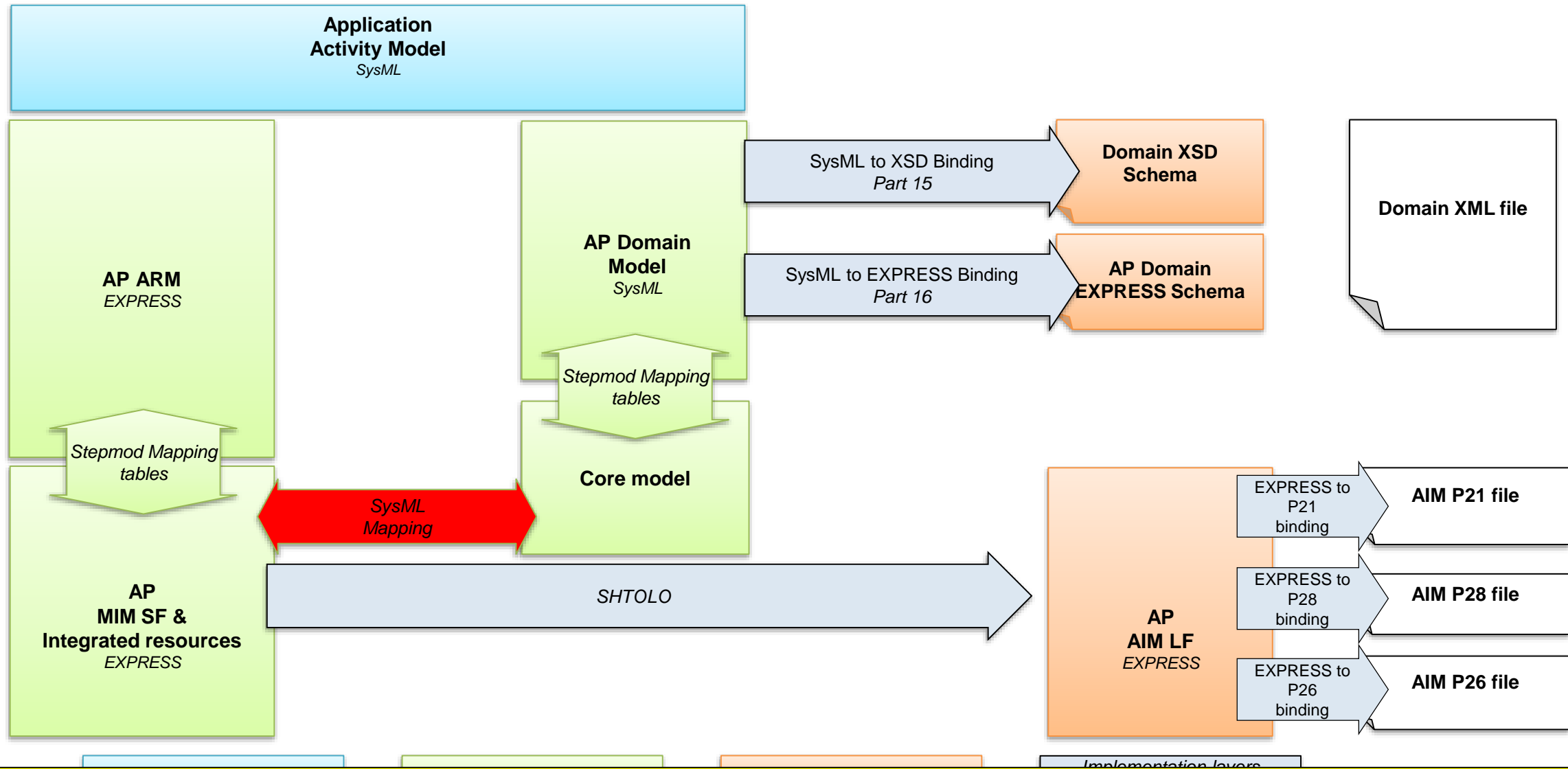


NIST



PDES, Inc.





The SysML Mapping between the Architecture extensions and the legacy modular infrastructure will allow to reduce the complexity

Proposed ISO 10303 Document Architecture after SMRL workshop with ISO CS

• As – Is

- Parts 1 is an overview of ISO 10303
- Parts 11 to 18 specify the description methods,
- Parts 21 to 28 specify the implementation methods,
- Parts 31 to 35 specify the conformance testing methodology and framework
- Parts 41 to 62 specify the integrated generic resources,
- Parts 101 to 113 specify the integrated application resources,
- Parts 201 to 242 specify the application protocols,
- Parts 301 to 332 specify the abstract test suites
- Parts 401 to 442 specify the application protocol modules,
- Parts 501 to 523 specify the application interpreted constructs
- Parts 1001 to 1999 specify the application modules.
- Parts 3001 to 3099 specify business object models,
- Part 4000 specifies the core model,
- Parts 4401 to 4499 specify domain models.

• To – Be

- Parts 1 is an overview of ISO 10303: (Simple Template)
- **Part 2** specifies the Vocabulary of ISO 10303 (XML, Simple Template)
- Parts 11 to 18 specify the description methods: (Simple Template)
- Parts 21 to 28 specify the implementation methods: (Simple Template)
- Parts 201 to 242 specify the application protocols: (HTML Packaged with 44x parts and Part 444x integrated)
- Part 1000 specifies the STEP Resource Library (SRL) to consume Parts 41-113, Parts 501-523 and Part 4000: (**Model Based managed in Jira/GIT**)
- Note:
 - Retire parts 31 – 35 and -301 – 332
 - **Retire parts 1001-1846**
 - Disposition 13584-20
 - Electronic Inserts continue to be published on standards.iso.org

We are streamlining the handover to ISO and publication process by transforming our modular document into a modular library managed as a code library.