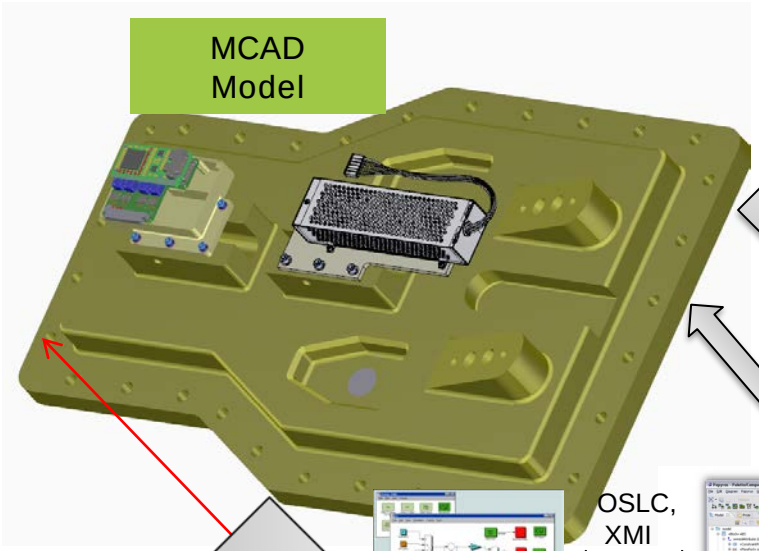
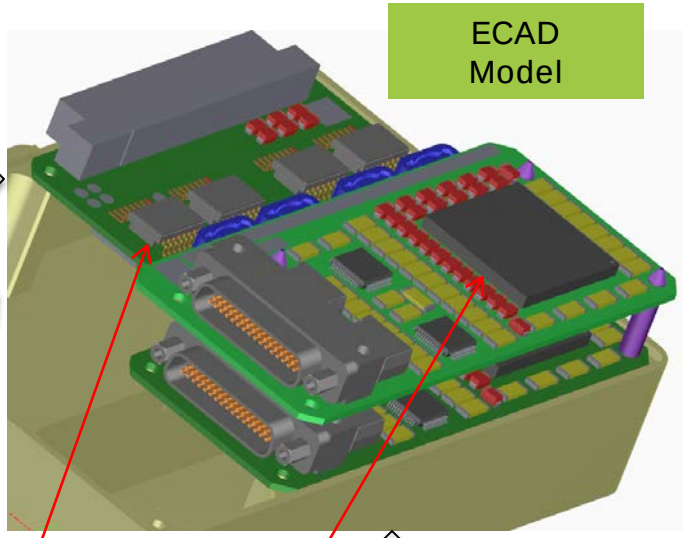
2017
annual INCOSE
international workshop
Los Angeles, CA, USA
January 28 - 31, 2017

PDES Requirements Traceability Project Update

Requirements Traceability Test Case

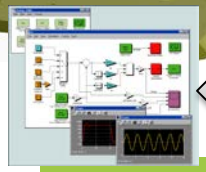


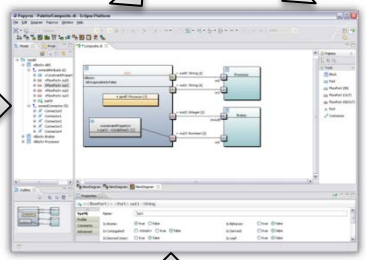
STEP
AP239/233/242/
AP203, QIF,
IDX, **PRC PDF**



OSLC, STEP
AP239/233/242,
PDF
EDX

OSLC,
XMI



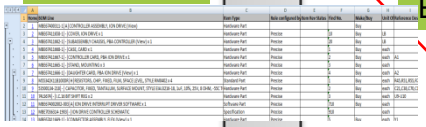


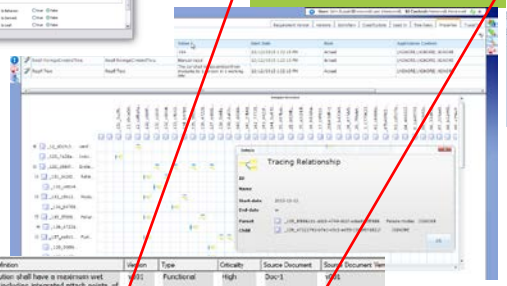
DBOM &
EBOM

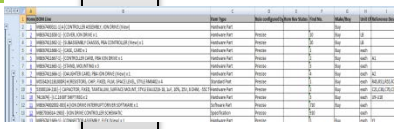
MBSE
Environment

Tracing
Management

DBOM &
EBOM



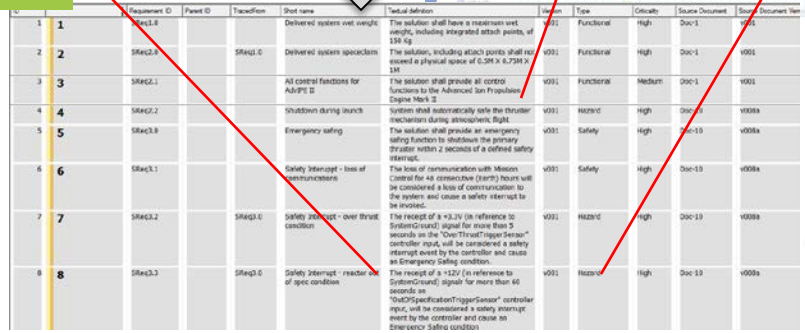




STEP
AP239/233/242/
AP203, QIF,
PRC PDF

OSLC, ReqIF,
XMI, PDF

Requirements
Management



STEP
AP239/233/242,
OSLC, **ReqIF**,
QIF, EDX, PDF

Requirements Traceability Status

Testing Environment

- **Share-A-space 7.8** Test Instance up and supporting: AP233/AP239, ReqIF (Eurostep)
- **Eurostep Nova In-Reach** – loading with requirements and tracing relationships, setting up for review – artifacts transmitted to Eurostep; more samples generated
- **DOORS 9.6 ReqIF** – loaded with updates from SME feedback
- **DOORS NextGen** – loading requirements; OLSC enabled but incomplete on trust relationships – ReqIF import from DOORS 9.6 successful
- **Enterprise Architect 12** – requirements imported through CSV; exported as XMI and transmitted to Eurostep – research shows EA12 can only act as provider; Oauth support looks questionable; next step is to work with Sparx
  - References of ModelSIM wrappers to provide OSLC interoperability
  - Feedback from others regarding difficulties with this version of application
- **Mentor Graphics Vx 1.2** update installed; both cards and flex tape converted. Conversion to **Vx 2.1** in process and access to better STEP, ODB++, EDX translators. Have 3D library available but haven't converted entire design yet.
  - In discussion with Mentor about a Context SDM test instance with OSLC
- **Creo 2** – Have parts modeled with PMI; incomplete conversion to fully public models; PTC willing to convert to STEP AP242 in Creo 4 preliminary version – Pursuing conversion to Creo 4; seeking version with AP242 translator
- **NX** – models not as well done as Creo (particularly for PBA's); available in NX 11; have access to preliminary STEP AP242 translator

Requirements Traceability Status - 2

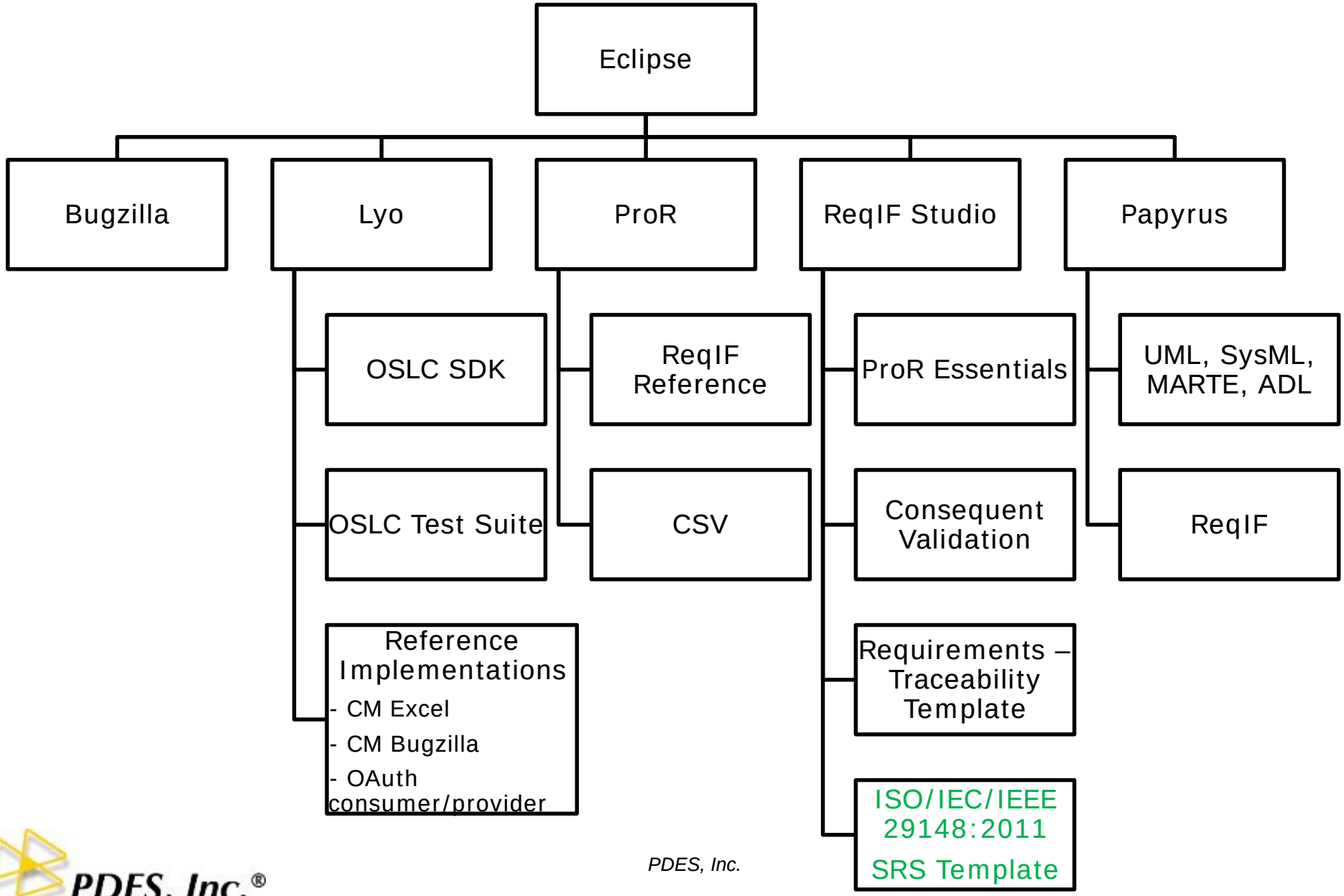
- **ARAS OSLC** – IBM releasing OSLC connector for ARAS (paid model) in 2017.
  - Would prefer native support within Aras for OSLC, but not committed
- **PRO-R** – updated to ReqIF Studio, new ReqIF import generated from DOORS.
  - Downloaded Eclipse – Lyo; working through setup.
    - Includes interactive and command line validator for ReqIF syntax
- **DoorScope** – installed and testing; did well with DOORS ReqIF but not reference files
- **Papyrus** – installed and testing; developing requirements diagram.
  - Unsuccessful (to date) on getting XMI import / export working
- **Tc 10** – EBOM structure loaded; product structure loaded with both a Creo 2 version and NX 8.5 version. Reference documents and PDF visualizations produced and linked to EBOM structure. PLM/XML output successfully exported and imported to other test instances
  - Should be able to produce ASME BOM reports shortly – first ones available but need to make generic
  - More test parts and public documents loaded
- **Clear Case** – access enabled for use of test instance with OSLC function; delayed on build software example
- **ANARK** – Working with latest versions and comment extraction function.
  - 1ST iteration of merged requirements, external attributes, and CAD data went well.
  - Need feedback on next steps

Requirements Traceability Status - 3

Test Data

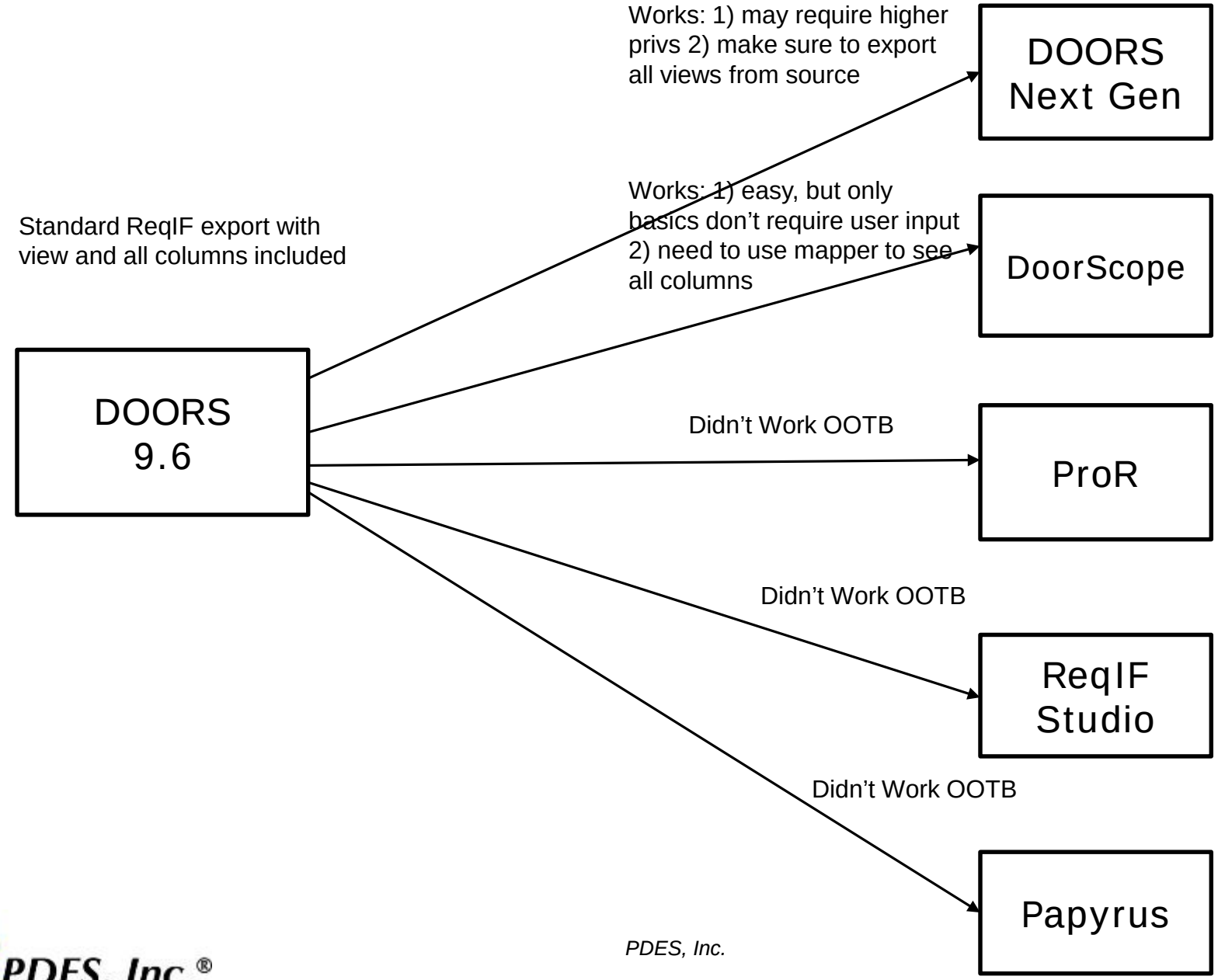
- ✓ Refresh Mechanical Product Structure – completed for Creo 2 & NX 8.5
  - Verify with NX 11 – open
- ✓ Ported Mentor Expedition examples to Mentor Vx 1.2
  - ✓ Add additional trace links to requirements in tools – completed
  - Investigate GUID links to requirements
  - Migrate to Vx 2.1 and xDM 2 – 60% complete
- ✓ PLM (DBOM) & PDM (EBOM) product structures – completed
- Exchange XMI versions with Enterprise Architect and Eurostep tools – in progress
- ✓ Build out software example – complete in PLM/PDM environment but need re-verification
- Build out Clear Case test cases – not started
- Update mechanical models for feature set use cases – not started
- Enter requirements into new version of Share-A-space – 25% complete
- Generic version of mechanical design – 3rd party made progress – 50% complete
- ✓ Updating requirements/traces for better configuration management tracing – complete, but concluding that including GUID's early in the process would help most exchange scenarios
- Obtain more analysis artifact examples – progress with Open Modelica and Visio timing diagram module

Eclipse Resources: ReqIF & OSLC



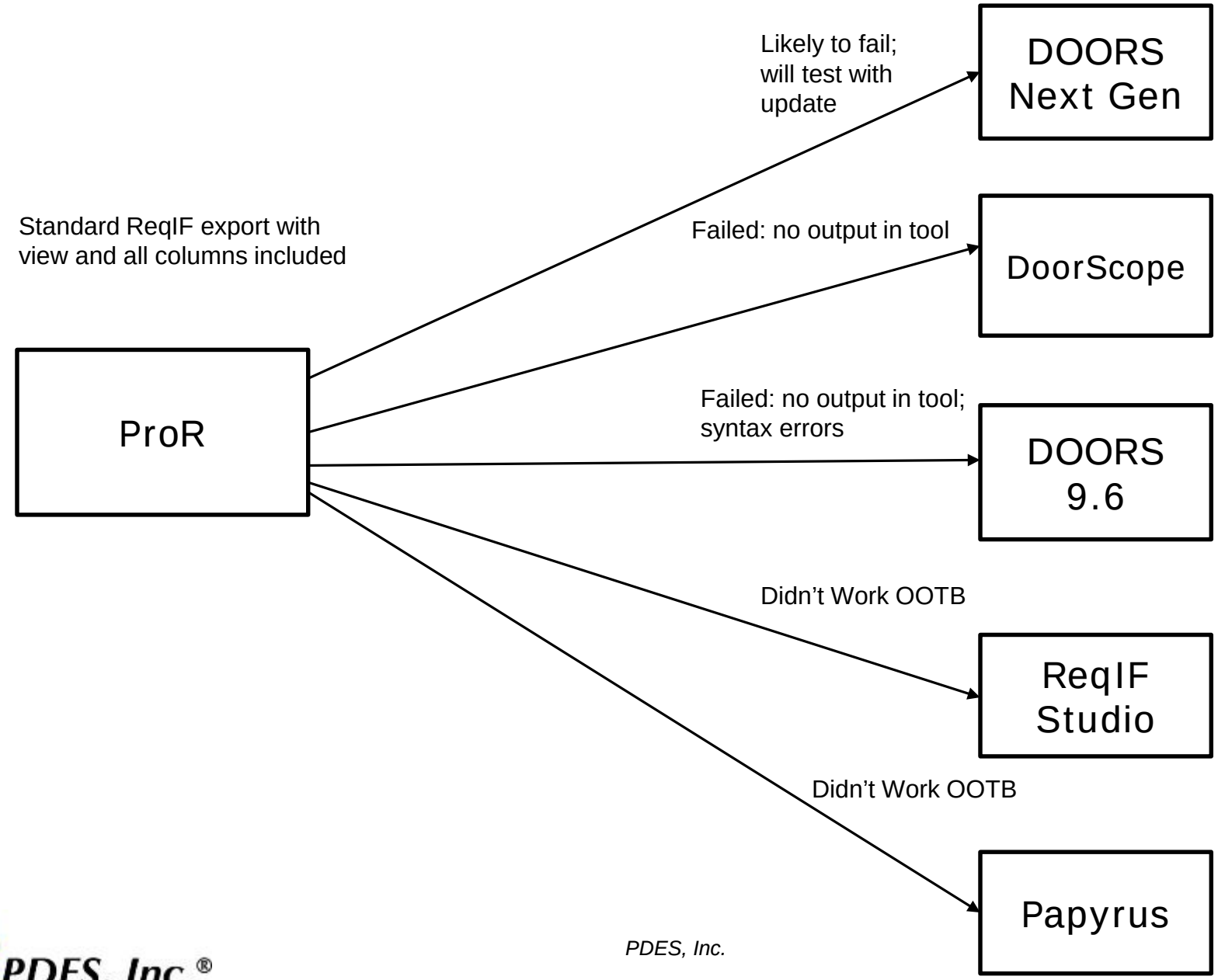
PDES, Inc.®
Connecting the Digital Enterprise

ReqIF – 12/6/16



PDES, Inc.®
Connecting the Digital Enterprise

ReqIF – 12/6/16


Likely to fail;
will test with
update
DOORS
Next Gen
Standard ReqIF export with
view and all columns included
Failed: no output in tool
DoorScope
ProR
Failed: no output in tool;
syntax errors
DOORS
9.6
Didn't Work OOTB
ReqIF
Studio
Didn't Work OOTB
Papyrus

Airbus STEP AP242e1 Requirements Test Case

Examples of information coming from the level 1 requirements file:

-**Identifier**: Unique identifier for the object filled automatically by DOORS RMF

-**Requirement Version**: Version number of the requirement that enables to manage requirements into configuration.

-**Requirement Statement**: States an expected behavior or global performance of the product under consideration.

-**Working Status**: Current version of the requirement/activity.

-**Rationale**: Provides the justification and/or the reason for the activity/requirement. The rationale is particularly useful for orphan requirement (i.e. requirement that are not traced to an upper level requirement).

Manually generated with XML tools

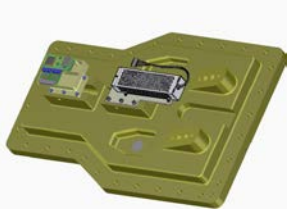
- 10-15 requirements
- No V&V elements
- Some differences in tracing relationships with AP239/233

Public release accomplished

- Downloaded and planning to incorporate
- Exploring possibilities for creating physical artifacts to go with AP242 example

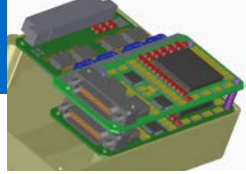
Contact: Claude Reyterou (Airbus)

MCAD Model Considerations



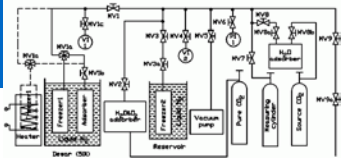
- MBE Offers Major Advantages Over Drawing Based For Tracing
  - Drawings and models don't need to match in traditional approaches
  - Models are normally parametric, drawings are not
  - Assemblies and derivative designs often contain mixed modes
- Use Case Requirements
  - GUID / UUID for features and feature groups (surfaces, dimensions, parameter-driven elements, notes, holes, etc.)
  - Pull from model: linked references to features and collections of features that are stable throughout the life of the model
  - Push to model: requirements, CTQ's, and constraints from MBSE into MBE models that can be exchanged
- Challenges
  - Inconsistent MBE modelling practices – many part model and especially assemblies not going through adequate V&V processes
  - Mix of MBE and legacy drawing approaches in large assemblies
  - Translation functions for textual requirements to usefully relate to features in MBE model – implementations not supporting all necessary features yet

ECAD Model Considerations



- Modern and Legacy Designs Support Logical Component Tracing
  - Tools tend to support hierarchy well over last 25 years but translations don't always handle well
  - Common reporting functions down to component pin level or even gate and internal package – easy to pull during baselines
  - Rules engines enforce design, producibility, and consistency checks
- Use Case Requirements
  - GUID / UUID for component.pin structure, signal names, (surfaces, dimensions, parameter-driven elements, notes, holes, etc.)
  - Pull from model: linked references to features and collections of features that are stable throughout the life of the model
  - Push to model: requirements, CTQ's, and constraints from MBSE into MBE models that can be exchanged
- Challenges
  - Mechanical features that are not components or in the ECAD library will be treated differently
  - Navigating hierarchy could be complex – particularly with programmed devices, ASICS, FPGA's and other components developed through external toolkits.

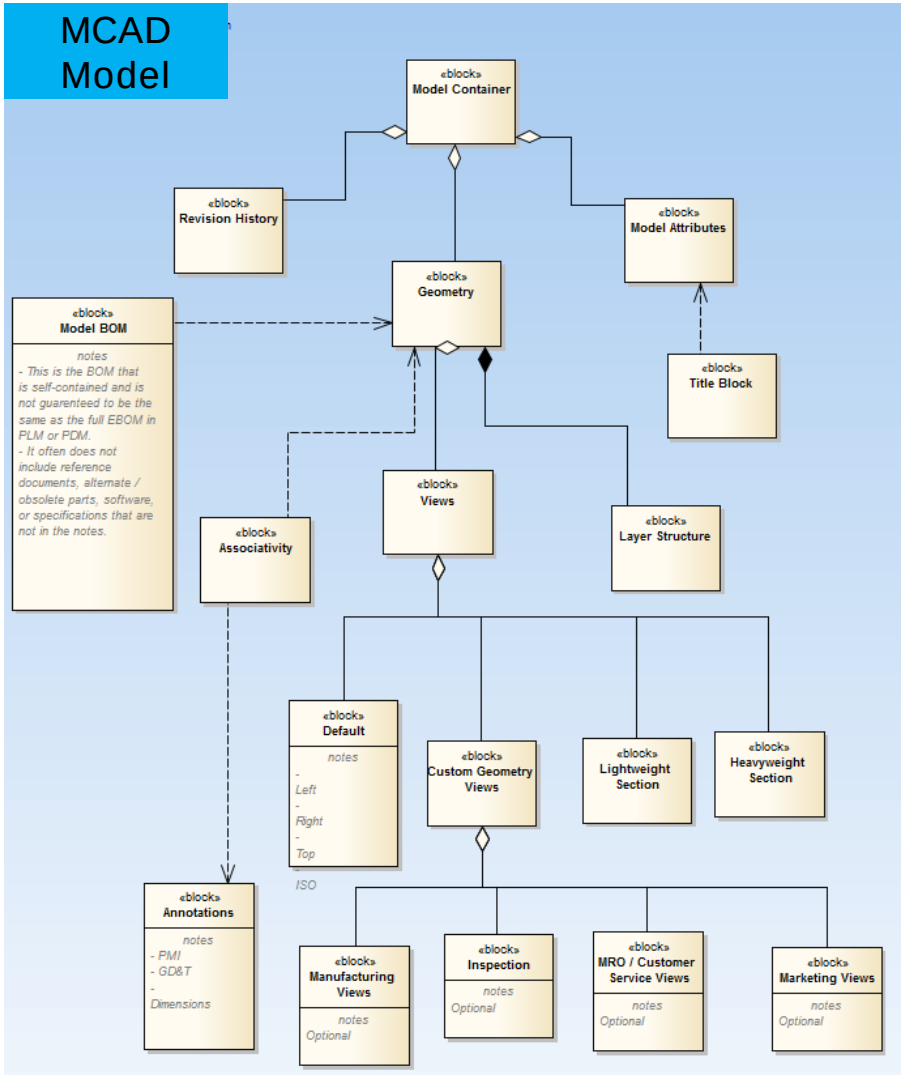
Schematic Diagram Considerations



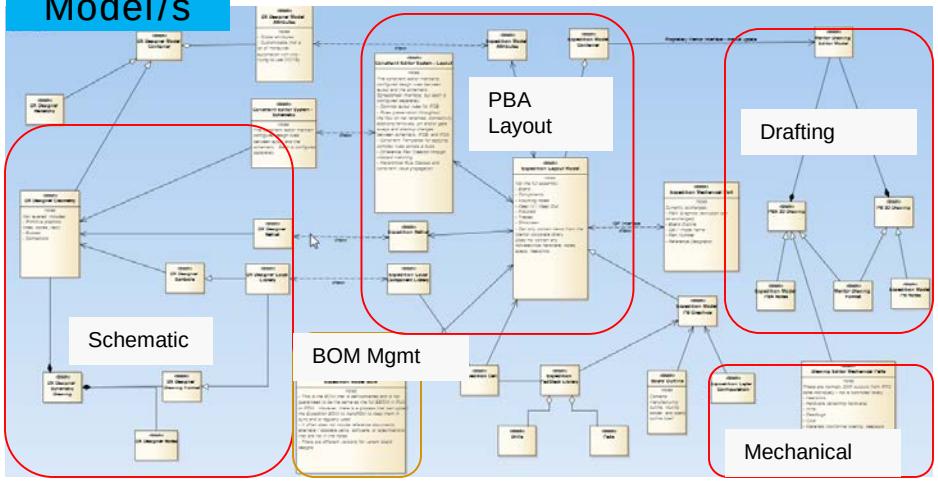
- Many Different Types
  - Tools tend to be domain centric (e.g. ECAD, computer logical, mechanical, etc.)
  - Basic graphics exchanges work well, but logic content exchange is poor or non-existent.
  - Hierarchy often not handled well
- Use Case Requirements
  - GUID / UUID for component.port structure, signal names, flows, etc.
  - Pull from model: linked references to features and collections of features that are stable throughout the life of the model
  - Push to model: requirements, CTQ's, and constraints from MBSE into MBE models that can be exchanged
- Challenges
  - features that are not components or in the library may be treated differently
  - Navigating hierarchy sometimes complex – particularly with components integrated through 3rd party toolkits and vendor libraries.
  - Representing bus, bundle, and manifold structures
  - Schematic tools for mechanical and electrical systems are not often the same

MCAD vs. ECAD Model Representations

MCAD Model



ECAD Model/s



ECAD design operates as multiple tools integrated with a local database while MCAD design models are often self contained models

ECAD design for a single part number can incorporate tools from different vendors to create a native model. The designs are normally dependent on a central library for physical, logical, and analysis models.

ECAD Designers are often tasked with maintaining configuration files to successfully manage their designs. **Collaboration needs to be concerned about synchronization of a lot of files in many formats.**

MCAD vs. ECAD Collaboration Differences

MCAD

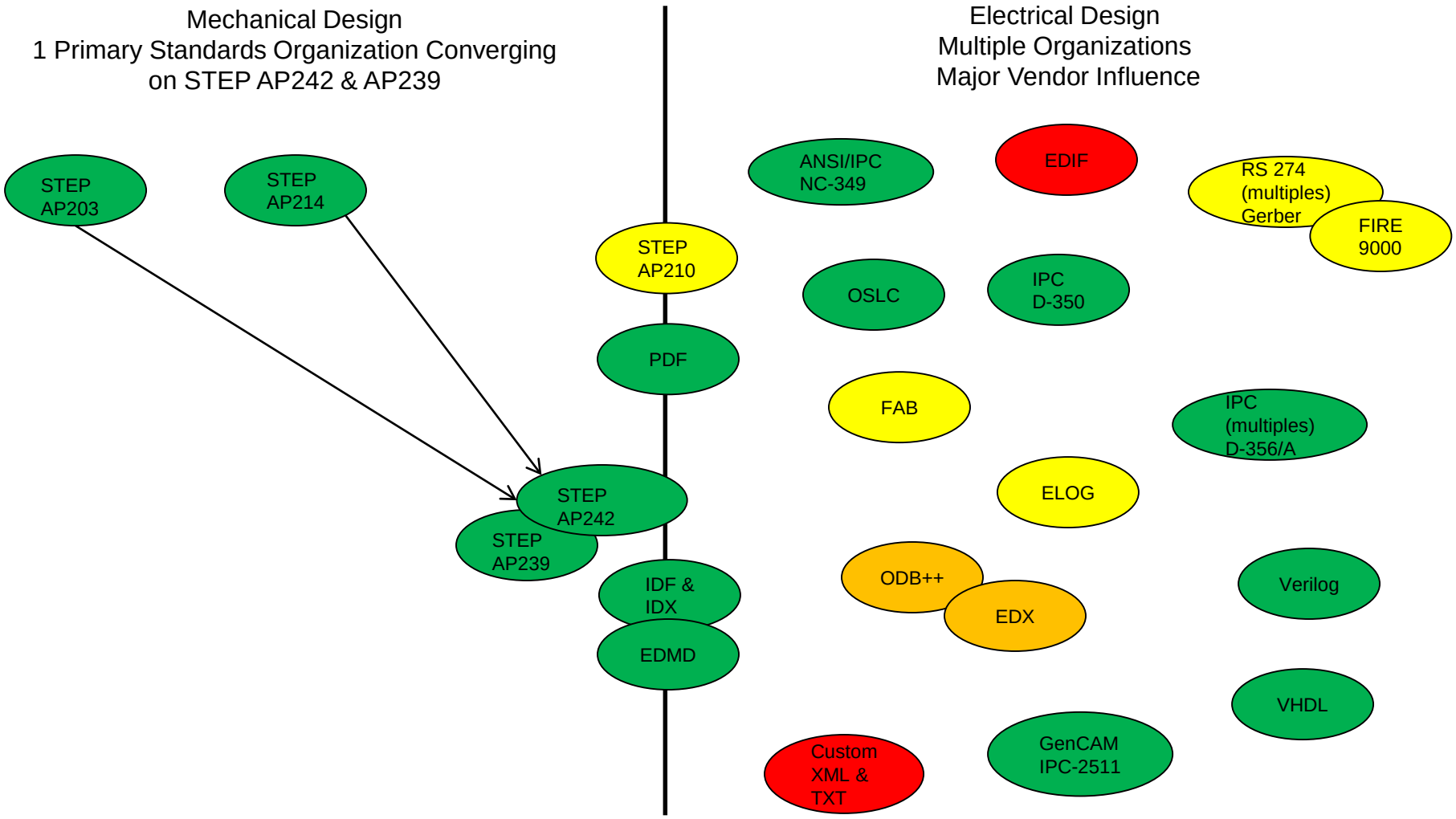
- Modeling done in single tool; supply chain has common view throughout process
- Components are less of the content and have small amount of alternates.
- Piece part activities – even in assemblies – are focus of design and supply chain collaboration. On-drawing partslists still popular.

ECAD

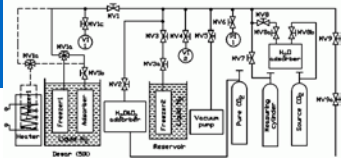
- Design process can be partitioned many ways among different tools but end result must reside in central PLM or library-connected repository
- Alternate parts, altered parts, and Source Control Documents are common. BOM primarily based on commodity items.
- Almost everything is an assembly – on drawing partslists don't work well

PLM Systems Traditionally Oriented Towards MCAD – That's Changing

Major Mechanical and Electrical CAx Standards

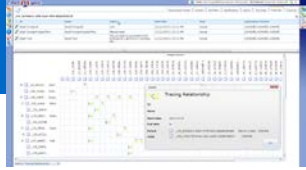


Document & Spreadsheet Considerations



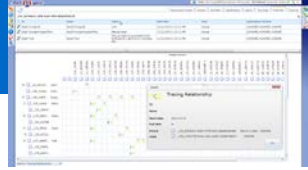
- Lack of Consistency
  - Project-developed macros and structures – some linked to other custom artifacts and databases that may not have documented schemas.
- Use Case Requirements
  - GUID / UUID for component.port structure, signal names, preliminary partslists and alternates, costs
  - Common for early development and reuse of existing designs
- Challenges
  - Maintaining consistency through lifecycle so every update doesn't turn into a new translation and mapping project
  - Lack of defined, basic common templates that could be shipped with major tools
    - These could be out there, but not commonly recognized – is this a potential activity?

PLM System Considerations



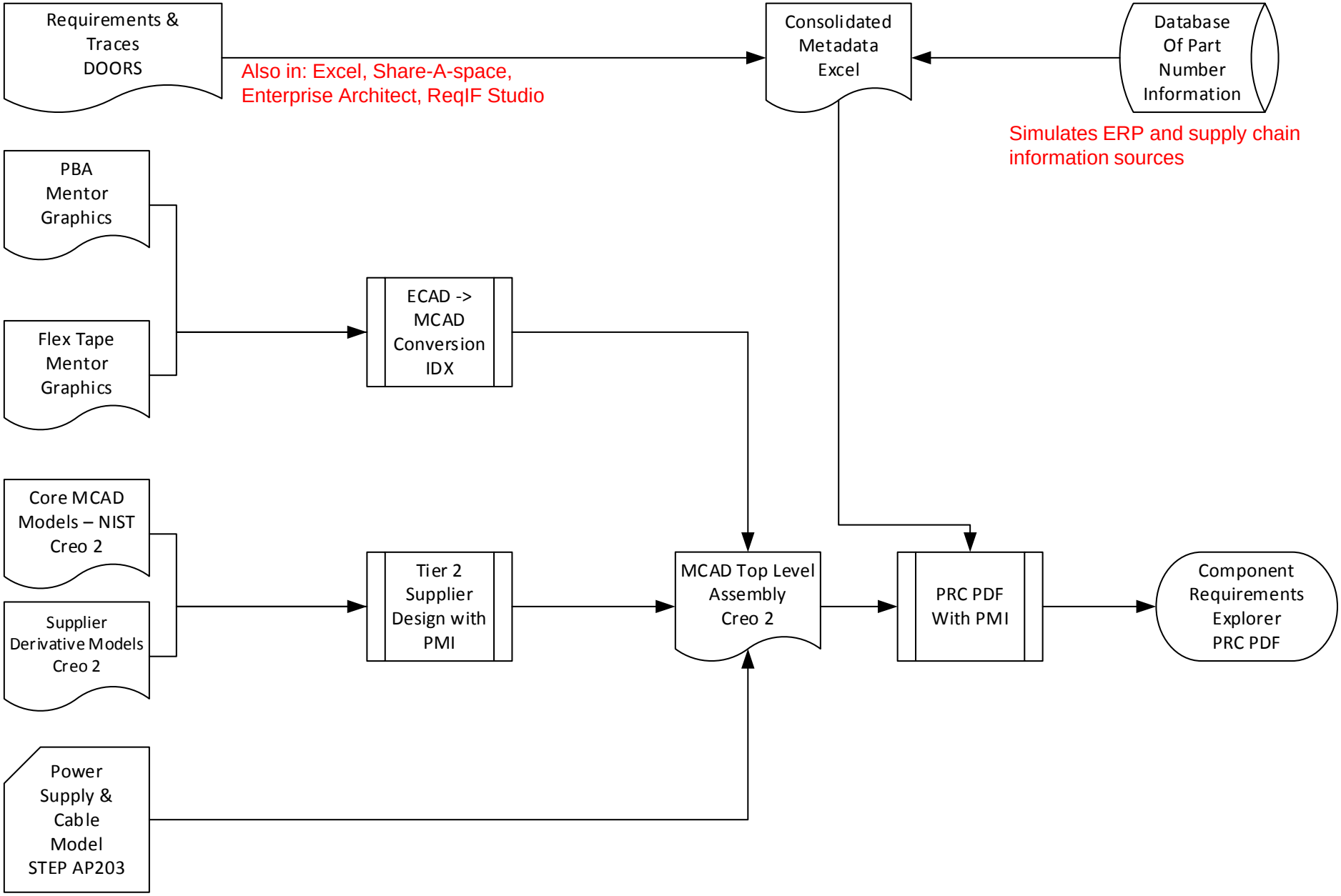
- Role of the PLM system in providing traceability
  - Primary objects: xBOM structures, reports, CAD models, schematics, change management objects, metadata status of objects
  - In an active environment that supports OSLC, what functionality does the PLM system offer when interacting with the models, artifacts, and metadata objects in the repository?
  - Should each artifact type (e.g. document, spreadsheet, MCAD model, schematic, ECAD model, etc.) be treated like a software language class so it could be addressed either by PLM or independently? This may entail a standards based API approach: AP239/233 representation of element relationships
  - Provides IP control, status, and one-stop shopping for hardware and product structure information
- Challenges
  - There is tremendous potential for PLM workflows, translators, and services to coordinate controlled access to CAD and cost relationships, but inconsistencies can thwart automation
  - During development, it is critical to classify CAx artifacts that may be in progress

Visualization Findings

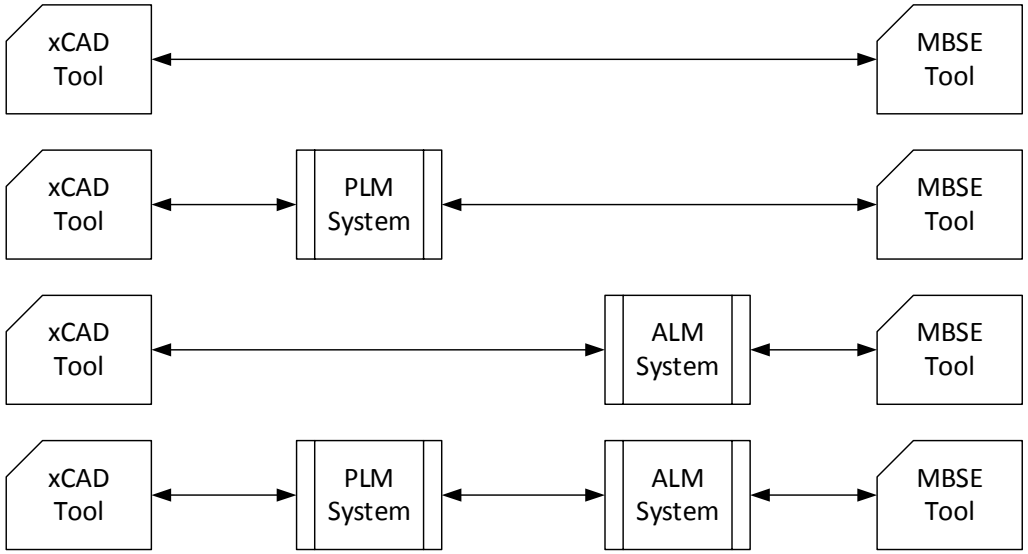


- Visualization Useful for Physical Products
  - Can be static, manipulated, or animations
  - Best if reviewers and stakeholders can capture comments in viewing environment and have it compiled automatically.
  - Growing number of tools can do basic operations OOTB
  - Limited use for software; particularly with interactive animations
- Associating Multiple Requirements to 1 Object Not Always Supported
  - Many systems oriented towards display of block of text and part attributes vs. accessing individual requirements in a controlled fashion
  - Can be done, but requires some interface planning to be useful
- HTML-based Approach Preferred Internally
  - Easily integrated with PLM, ALM, ERP resources
  - Security an issue for people outside the organization: handling links to IP sensitive resources, account management, partitioning interactions and feedback of competitors, etc.
- File-based Artifact Exchange Approach Preferred Externally
  - Current systems find this easier to distribute and manage
  - Can be distributed to downstream suppliers – both good and bad with that that.

Visualization of Tracing Relationships POC



Scenarios Considered



- Scenarios Change During Lifecycle
- Tool-only Scenarios Are Difficult to Track and do IP Control
  - Domain differences and common references need to be understood for meaningful results
  - Improvements in exchange standards requires better alignment with high priority use cases
- System-based Scenarios Are Difficult To Configure And Maintain
  - May not happen as often as they should
  - Standards are more complex but more complete

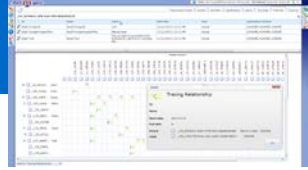
Use Cases – ALM View

- Status Reporting & Queries
  - Links between ALM and PLM environments to enable real-time reporting on PLM objects that implement requirements
  - Real time access to program status
  - Ad-hoc rollups for Systems Engineering on PLM, ERP, and historical performance related to product structure
- Impact Analysis and Validation
  - Program risk accurately assessed with more automation
  - I/O validation and tracing
  - Workflows include ALM and PLM objects with increased granularity
- Reuse
  - Search for current and historical elements linked to product or platform structure
  - Expert identification
  - Functional block reuse and history on trade studies
    - Updates to PLM and hardware toolsuites improving reuse

Use Cases – PLM View

- Status Reporting & Queries
  - Links between ALM and PLM environments to enable real-time and historical reporting on ALM objects that drive design constraints
  - Identification of analysis results not in PLM; eliminate investigations that were already rejected in trade studies
- Reuse and Impact Analysis
  - Program risk accurately assessed with more automation
  - Workflows can included ALM and PLM objects with more granularity
  - Access to studies to aid in cost reduction
  - Analysis date to improve test and inspection processes

ALM-PLM System Interoperability



- ALM – PLM Exchanges
  - Metadata is easy; artifact features are difficult once we go past basic access to individual files
  - Building multiple views of linked data
  - Baseline and archival at project level
  - Exposure of MBSE activities and status to PLM users.
- Challenges
  - There are a huge amount of relationships that can be established that would need to be maintained:
    - Prioritizing use cases for business and product impact
    - IP controls – easy to accidently expose information to customers and suppliers that is inappropriate
    - Persistence and archival of relationships, artifacts
    - Legal ramifications of object relationship archival

Project Summary

- ALM – PLM Exchanges
  - Metadata is easy; artifact features are difficult once we go past basic access to individual files
- MCAD and ECAD Tracing to PMI or Model Features Possible …. If Model Based Design Practices Are Followed
  - Drawings Are NOT Models
- Publicly Available Test Cases Were Critical to Project
  - Need work on high priority use cases and test elements for software
  - Control of test artifacts by central organization preferred
- Maintaining Detailed Tracing Throughout Lifecycle Requires ALM and PLM Disciplne
  - Scenarios critical to business and process must be identified and supported for PMI traces – doesn't happen automatically (yet)
  - ALM <-> PLM likely to work more reliably than tool to tool situations

Discussion Topics

- How relevant is this project to your Systems Engineering practices or needs?
  - What would improve it?
- What examples would people like access to?
- Are there projects this activity should align with?
  - How could we pull this together into perhaps an MBSE community of practice?
  - Could INCOSE, NIST or some other entity host and inventory / control the artifacts?
- For useful software example, what could/should we build?
  - Is anyone willing to help with that? - LOTAR and MoSSEC efforts could benefit from that
  - Are there standards for user interface prototyping or system interaction applications we should consider looking into / supporting?
- SAVI did some excellent work. Is there some way, we could build out / replicate a public version of the SAVI examples?

Discussion Topics II

- What scenarios are people interested in and how do we trace down to potential artifacts that support them?
  - Then what derivative requirements are there on the artifacts and relationships so they support the scenarios?
- Missing in this are details around some standard, basic office automation tool templates commonly used in Systems Engineering, Requirements Management, and inter-company exchanges on large programs. Would it be of value to put out some pseudo-standard templates with filled out examples for:
  - Requirements
  - Tracing Matrix
  - EBOM's
  - Interconnect Tables / Wiring Lists

2017

annual INCOSE international workshop

Los Angeles, CA, USA

January 28 - 31, 2017

www.incose.org/IW2017