Improving Tools and Processes in Mechanical Design Collaboration

Clark Briggs[*]
Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109

Cooperative product development projects in the aerospace and defense industry are held hostage to high cost and risk due to poor alignment of collaborative design tools and processes. This impasse can be broken if companies will jointly develop implementation approaches and practices in support of high value working arrangements. The current tools can be used to better advantage in many situations and there is reason for optimism that tool vendors will provide significant support.

Nomenclature

A&D	Aerospace and Defense Industry
BOM	Bill of Material
CAD	Computer Aided Design
CAGE	Commercial and Government Entity
COTS	Commercial Off The Shelf
IPDT	Integrated Product Development Team
IT	Information Technology
PDM	Product Data Management
PLM	Product Lifecycle Management
STEP	Standard for the Exchange of Product Model Data, ISO 10303
WAN	Wide Area Network

I. Introduction

COMPANIES in the aerospace and defense (A&D) industry have a high value business need to significantly advance their collaborative product development capabilities. This paper calls for cooperative development and demonstration of implementation recipes. This is based upon the following foundation:

- Current business practices (e.g., contracts, design authority) mean that read-only access to accurate geometry is sufficient to radically improve the efficiency of collaborative mechanical design.
- STEP provides sufficient (and excellent) neutral standards for configuration managed exchange of CAD geometry.
- The A&D industry needs deployment recipes (validated and published) for a few high value heterogeneous PDM and heterogeneous CAD combinations.
- Process mismatch in current A&D organizations must be addressed in order to take advantage of new tool implementations.
- Joint cooperative demonstration of model exchange in collaborative design use cases is within reach of A&D companies.

II. The Nature of Development Work in Aerospace & Defense

Collaborative product development in A&D here means the joint development of an aircraft or spacecraft by multiple organizations. The companies are typically a system integrator and one or more first tier suppliers. Their engineering work teams are usually organized in Integrated Product Development Teams (IPDT). These teams are more often than not widely distributed around the globe.

Although they work cooperatively, the work is partitioned either by subsystem or by physical sector of the aircraft. The system integrator is responsible for putting the contributions together, flowing requirements down to the suppliers and coordinating the progress of the ensemble of teams. The suppliers are responsible for their major components or subsystem, providing designs that meet the higher level requirements. Current project management

[*] Hardware Development Process Engineer, M/S 158-224, and Associate Fellow.

practices coordinate this division of the design activity by defining and managing the interfaces between the suppliers' subsystems.

Despite the appearance of strict division of labor, the concurrent nature of IPDTs and the compressed business cycle times mean that the teams collaborate intensely. Collaboration has many components, such as virtual meetings and reviews, give and take of design trades across company and contract boundaries, and common integrated electronic data systems. In this paper, the focus will be on the joint utilization of the system design (mechanical CAD) models, management of and configuration control of the system models and the exchange of models representing changes to the system design.

The system integrator might have significant development activities beyond system definition, integration and oversight of suppliers.

Trades and design choices are made across these interfaces as well as inside a single organization's activities.

The nature of work agreements in A&D is based upon dividing responsibility. This is often formalized as Design Authority. This business structure is reflected in the contracts which are formulated and put in place before work begins and, as a result, must forecast all the exchanges and collaborative interactions. This division and formalization in contracts means that a single entity has ownership of the design, responsibility for the design's performance and authorship of the CAD models. Even inside a given organization, good project management practices assign the responsibility to a single work unit or team which usually has a single responsible person leading that unit.

Therefore, with few exceptions, the design is authored, written, decided by a single entity. In A&D today, the design authority pursues these responsibilities in collaboration with and in consultation with others, but the decision maker is a single known entity.

(A common exception occurs at interfaces between responsible units. Frequently one side of the interface will propose a design change by modifying the other's design in the CAD models simply because that is expedient and it is too cumbersome to make up/craft/design a joint solution in these cases with each only modifying their own CAD models. Still, the changes are proposed to the other for acceptance.)

The acquisition and consolidation in the industry has resulted in some progress toward standardization of tools and processes within larger companies, but substantial internal alignment remains. Across the A&D companies, there is no business value today in changing tools to better match external business partners. For system integrators, this is true by supposition. For tier one and two suppliers, this is demonstrated by simple economics. The market place currently provides, for all intents and purposes, a balanced field of collaborative mechanical design tools. No one tool vendor has sufficient widely understood technical superiority or market share to motivate any company to retool in their products.

Therefore, a basic premise of this paper is that changing tools is not an option and somehow the ability to work together collaboratively must be developed on top of the current tools.

III. Interoperability

As a result, interoperability takes on the minimalist meaning of supplying a sufficiently complete design context for a unit's work. Geographically neighboring designs or relevant designs authored by others simply need to be present with sufficient fidelity but need not be modifiable.

(Fully parametric models are not required for most A&D interchanges, since the receiver does not possess the design authority to change or modify the originator's models. Removing the parametrics while retaining the geometry results in "dumb models.")

Vendor neutral standards, STEP specifically, are excellent for this purpose. High accuracy, dumb solids are straight forward to supply to the others for their design context.

A second premise of this paper is that collaborative design is significantly facilitated when all participants have a coherent coordinated common system model. Access is subject to rights. STEP AP203 is the way to submit segments of the design content for accretion into the system model.

IV. Configuration Controlled Exchanges

In the rapid pace of today's collaborative projects, the system design changes quickly and, to avoid excessive technical risk of error, the development partners must ensure that the design CAD model is a sufficient representation. A few years ago, projects might have been content with 2 or 3 annual updates to the system model. In today's environment, monthly updates are a desired minimum, but aren't widely felt to be achievable. (A few automotive industry reports cite weekly syncs and at least one global A&D project cites a weekly sync.)

Here again, STEP (AP214) provides the version control information for repeated exchange of CAD design models.

V. Use Cases

In anticipation of developing and demonstrating standard practices for collaborative exchange of system models, the following use cases are presented. They abstract the activities that working organizations will need to be able to carry out easily and often. The first two illustrate preparing and updating the master system model while the third and fourth illustrate using the model in visual exploration and design. Further elaborations are provided in Appendix A.

A. Establish Common System Master Model

The first instance of the master system model needs to be constructed. As the first instance, new objects will need to be constructed and populated with the contributions from the development partners. Initial inconsistencies in naming and structure must be reconciled and the a priori rules for naming, submission and identification updated if required. Since this is the first look at the assembled geometry, design content issues will likely be present and a decision to fix them in this round or a subsequent update made.

B. Update System Master Model

With an existing master system model in place, the risk of technical errors increases as design decisions across the project are taken but not incorporated into everyone's model and distributed. The master model needs to be updated to capture the current state of the design and to reconcile the consequences of isolating participants' work behind interface definitions. The model maintainers might put out a call to the contributors that establishes the date and required content. Contributors finish up work on the current models, package up their contributions and transmit them to the maintainers. The maintainers prepare the next working version, fold the contributions in to make the updated model and reconcile changes. Simple errors might be fixed in coordination with a contributor, but it is to be expected that design inconsistencies will remain to be worked in the subsequent period. The updated system model is made available to all participants.

Updating the system master model requires a set of practices. For example, the reconciliation team would be significantly aided by a guide to the changes provided by the contributors. The individual models might have new names, the tree structure might have been rearranged, tree branches might have been deleted or new branches added. This modified tree must be laid on the existing tree as a new working revision and this is much easier to accomplish with a few hints from the originators. Naming conventions play an important role here in eliminating collisions. Broken names significantly increase the manual effort to update and ruin the version to version continuity. Contributors must submit only their work (with updated components and instructions for unchanged components) and must not return any elements of others' work in their possession.

The architecture of the project PDM also affects the update process. If the project keeps a single PDM for all first and some second tier suppliers, submission, reconciliation and update happen in a single system on a single copy of the CAD models. Suppliers are in a position to accomplish their portion of the issue resolution and reconciliation. The update might be laid out in phases. As the planned date approaches, the contributors finish up their work and prepare for the update. The reconciliation team assesses the draft updated system model and proceeds with issue resolution and fix up. Upon satisfactory completion, the assembled and repaired model is made available to all participants. Even though contributors might be in different locations, this process happens in the system PDM and for all intents and purposes, the models are not moved or transmitted.

If the suppliers maintain their work in their own PDM, the update progresses in similar phases. As the date approaches, the contributors finish, package up and transmit their submissions to the project model maintainers in an initial update phase. In phase two, the maintainers assemble the contributions in the master model PDM and perform reconciliations in consultation with the contributors. In pursuit of reconciliation, the contributors might provide additional submissions. In the final phase, the updated model is made available to or transmitted to the project suppliers who update the copy in their PDM.

The access rules must be applied in the update process. By presumption, the system integrator has access rights to the entire system model, but suppliers may be partitioned. Suppliers have the necessary access rights to bundle and submit their contributions from their work environment. The reconciliation team may well contain representatives of all suppliers as well as the system integrator and these supplier representatives must abide by the access rules while assembling the system model update. Finally, the redistribution phase must abide by access rules and correctly distribute the relevant portions of the updated system model to the suppliers.

For a number of reasons, the system model may not contain the complete definitive design expression in the suppliers' CAD models. Current tool limitations, for example, provide practical limits to the size of the assembled models. Agreements to protect intellectual property may impact the modeling process differently than access rules based on security. As a result, contributors might submit reduced order representations based upon, for example, the outer hull of the design solids, a truncated assembly BOM, or elided design features.

As examples, an aircraft integrator might not receive the internal CAD model details of the engines or the spacecraft integrator might not receive the design details of an instrument. This might be pursuant to intellectual property agreements or concessions to model size in recognition that system level issues can be divorced from such internal details.

The result is that, prior to bundling up submissions, the contributors must first construct and prepare these alternate representations. With today's tools and work practices, this constitutes notable effort and the labor and calendar time for the update must be called out in the business activities captured in the contracts.

C. Review System Master Model

A high value use of the master system model is to use it in support of a review. This use case provides the opportunity to demonstrate access control as well as collaboration with remote participants. The viewing activities might be simply to explore the design visually to understand its layout, to measure aspects such as clearance or volume collisions, or mark up of observations.

D. Supplier Design in System Context

Given the project organization based upon division of work, design authority and geographical dispersion, most of the design activity occurs at the suppliers' location and in the suppliers' CAD and, in some cases, PDM environment. The design work of the others is represented in the design context but is not to be modified. The supplier pursues resolution of design issues, moving the CAD model expression of the design toward the objectives set for the next update and noting changes to be requested in the controlled interfaces.

Application of access rules and procedures is also prominent in this use case. The system model context present for some suppliers may be abbreviated due to access restrictions.

Further activity in preparation for the update might include development of reduced representations. As noted above, these might be necessitated by limitations on the size of the system model or in response to intellectual property or security agreements. Preparation activities might proceed as follows.

As the submission date approaches, the design team focuses on finishing the required content and cleaning up the models. At the closing date, the submission team captures the models and isolates them from subsequent work until after the update. Necessary preparation work, such as pre-pending names with the organizational identity or changing the metadata or owner names, is accomplished. The models are translated to the required format such as STEP or the system integrators' CAD format, packaged, and transmitted. The submission team participates in the reconciliation and issue resolution process with the system integrator, which might require additional input from the design team. This increment is captured, bundled and submitted. When the updated system model is issued by the system integrator, the submission team extracts the work of other participants that provide the design context for their team and updates the local system model. This might involve rearranging or pruning the tree, renaming or altering names, or even removing their own content returned in the system model.

VI. Use Case Business Structure

The mix of business and contract arrangements in A&D leads to at least three situation contexts for these use cases. Each presents different challenges to establishing and maintaining a collaborative environment.

A. System Integrator and Tier One Suppliers

This work hierarchy has a simple structure. In most situations the system integrator is the primary host and controller of the master system model and the suppliers are the contributors. All parties participate in reconciliation and collaborative design. In many circumstances, all parties are allowed access to the entire system model, but more and more often, access rules lead to partial distribution to the suppliers. In this simple working arrangement, few conflicts arise in schemas, naming, or standards. In reality, this is a multi-tiered structure when tier two suppliers and others are considered.

B. Major Subsystem Supplier to Multiple Primes

This situation is more complex and looks at the prior context from the point of view of a tier one supplier engaged, as a business entity, with multiple system integrators. Maintaining separation, packaging contributions, and deploying with different standards for each prime are significant issues to be worked out.

C. In-House Multiple Site

In A&D, there are a number of larger companies with multiple locations that contribute to a given project. In this business situation, a common company standard for practices and conventions may be in place, but not always. Access control by company employees is not a driving issue as it is with the other contexts.

VII. Practicalities

Beyond using STEP, the project participants need to decide a priori on certain practices. Conventions for naming and part numbering, mentioned above, is a well known example. To assemble a master system model from the contributions of the project participants, a numbering scheme that avoids collisions is needed. Most companies use their numbering scheme for parts for which they are the design authority, that they release and that they deliver as an assembled unit. Some parts suppliers are given part numbers from the buying organization. When subsystem models are assembled into the master design model, a common requirement of data management systems is that these all need unique numbers. One practical approach is to pre-pend the company's CAGE code and another is to cover all supplier parts with a part number from the system integrator. There are undesirable side effects of each of these approaches.

Beyond those, standard approaches and guidelines for other data are needed and the standards organizations and vendors in the tool market place aren't supplying widely adopted solutions. Personnel identities need a portable representation. Certain design properties such as material and/or density are needed and allowable values determined. A revision numbering sequence must be adopted, in particular prior to first release (Rev A) and between subsequent revisions such as Rev B. Once these conventions are selected, the data can be transmitted in the STEP files.

An efficient and successful export and import capability depends on good model checking. An automated, rule-based checking tool must check for geometry quality, required metadata, and naming. This significantly reduces manual effort in exchanges and is essential to resolving issues raised during exchange. Each contributing organization would deploy a checking tool and use it in the closing days leading to a submission. The reconciliation team might even require submission of the results to help them in assembling the system model update. Of course, continuous use of the checking tool between submissions is considered a best practice. The check rules are an excellent manifestation of agreed upon practices and must be established as part of the project start up activities.

In practice, the interim design changes made at an organization won't necessarily mesh with the changed design of another organization. For each update sweep of the design model, a reconciliation subcycle is needed to get an appropriately, but not completely, consistent design model. The reconciliation team needs to deal with changes in the model tree, appearance of new parts, and unexpected unconnected subtrees. With that completed, there is a single set of models that represents the current design iterate in a single place. This model must be distributed to all relevant participants who then transplant this update as the new working system model.

Here, the PDM architecture has a significant impact on model updates. With a single project PDM capable of supporting the system integrator, tier one and tier two suppliers, no models are transmitted, the model owners can fix up their own submission and meshing of designs can be monitored on a continuing basis. A single PDM might not fit the business relationship of the suppliers, the suppliers may have significantly different internal processes, or the project pace may not be so rushed as to warrant such closely integrated IPDTs.

When the project participants maintain their own design PDMs, automated submission of model updates may be possible if similar PDM tools are used. With sharply different PDM tools, there is presently little industry experience in automating transfers and best practice is to rely on manually prepared and transmitted submissions using STEP AP214.

VIII. Process Mismatch

While the finite set of A&D companies work together in relatively consistent arrangements, there are many new divisions and local process improvement activities within those divisions. As a result, the participants on any given project won't have well aligned design processes. In fact, tier one and two supplies may well find it impossible to align well at all with their multiple primes.

Beyond the practical issues of numbering and naming, significant process issues get in the way of regularly exchanging design models in a collaborative design activity. For example, a tier one supplier who works very formally by contract scope and deliverables might not be prepared to share interim design models with the system integrator in an update cycle if it wasn't negotiated into the contract up front. And, while the industry settles on the new collaborative design practices, these often aren't negotiated up front because of the ongoing process of discovery and improvement.

Another process mismatch might be the relative division of responsibilities between the technical design lead, the project engineer and the designer. Historically, CAD authoring has been such an intense and heavy activity that it required all of one engineer's attention to get it done. In other organizations, the engineer creates the CAD models as one component of their project engineering duties. If these two organizations are to work collaboratively together, each will finish up current work, box up the models, put someone's name on them and ship them to the reconciliation group, differently. The reconciliation team often finds this out on the fly when attempting to negotiate quick issue resolutions. The project engineer that built the contributed CAD models can readily agree to proposals, but often the designer can understand the modeling issue but can't agree to the content change without getting consent from the lead engineer. Trying to figure this out when the update deadline is approaching is painful.

Tools used across organizations differ widely. Beyond CAD and PDM management of CAD models, drawing release systems might be separate from the model management system. BOM management might be in an entirely different and separate system than release management or model management. Thus, one organization might find all the solid models and metadata in a single PDM system and a simple matter to package an iterate up for exchange. Another might find the models in one system and the BOM structure in another. The metadata might eventually be in the release system but prior to release it isn't anywhere. Building an exchange bundle would be a complex integration activity that would be difficult to repeat efficiently on a regular basis.

IX. A Categorization of Tools

Two basic classes of tools are required to accomplish configuration controlled exchange of design CAD models in a collaborative environment: the CAD tool and the PDM tool. Clearly, the CAD tool is the geometry authoring tool. The PDM tool is the configuration control tool for managing the CAD models.

It is commonly assumed that collaborative exchange would be simple if everyone used the same CAD tool, but this fails to recognize the need for a PDM for configuration control.

In the A&D industry, the current project participants, for a wide variety of reasons, have little to no alignment in BOTH CAD and PDM.

The phrase homogenous CAD describes the situation where all participants use the same CAD tool (and are on the sufficiently similar versions). Heterogeneous CAD describes the use of various CAD tools by the design entities.

Similarly, the term homogeneous PDM denotes that all project participants use the same PDM tool (and are on sufficiently similar versions). Then heterogeneous PDM means participants are using different PDM tools.

The nature of the tool market place currently is that there is not a single widely used CAD tool and its compatible PDM tool. We are richly blessed with 3 to 5 CAD tools that are entirely credible in some portion of the A&D context and 2 to 4 PDM tools that can credibly manage CAD models.

A project with homogeneous CAD and homogeneous PDM isn't interested in this discussion because they don't have these issues. The projects across the organization might not be homogeneous, though.

The least complex situation arises with homogeneous PDM and heterogeneous CAD. One or more viable implementations exist for this scenario, depending upon the particular tools. The master system model is then a single tree in the PDM tool that is populated with solid models from the various CAD tools. There are a few requirements to make this work practically. The PDM tool must really be CAD agnostic. A good visualization tool is essential to providing all participants a CAD-neutral view of the whole design. And, a good automated translation service is needed to supply each participants the system model context in their native CAD design environment. This situation is the simplest meaning of multi-CAD.

The most complex situation is heterogeneous PDM and heterogeneous CAD. Beyond a PDM handling multiple CAD models, the system master model must be contained in several different PDM systems. This makes the update reconciliation process and subsequent distribution of the updated master model extremely complex and cumbersome.

The situation of heterogeneous PDM and homogeneous CAD is not common, primarily due to the tortuous evolutionary path such collaborative partners would have to have taken.

Real businesses and projects in the industry exist in these different arrangements. As indicated, some situations are more tractable than others. Practices and approaches for the simpler ones will contribute to understanding and conquering the complex ones. The business value to finding solutions may differ depending upon the project and may thus more strongly motivate work in particular areas.

X. A Call for Implementation Recipes

A&D development organizations would benefit from working together and working with tool vendors to develop, demonstrate and publish how to build and use particular combinations of CAD and PDM tools successfully.

Tool vendors might improve efficiency and success of A&D development organizations if they published implementation recipes for their tools. Most tool vendors almost do this for their PDM and their CAD tool. In today's project environment this is not helpful very frequently. PDM tool vendors should help improve development organizations' capabilities by publishing recipes for heterogeneous CAD and homogeneous PDM using their PDM.

A&D organizations should establish joint projects to develop recipes for heterogeneous PDM lash ups. Although not currently popular, tool vendors could potentially be convinced to participate. Development companies could also nurture the maturation of tool-vendor-independent third party companies in the new market niche of PDM-to-PDM interoperability or data exchange.

The recipe producing activities would need testbeds to work out the recipes and demonstrate their readiness. A testbed would need enough hardware to establish the topology and should probably include one or more WAN links. The tool venders usually provide software and licenses for testing. A tougher issue is obtaining and loading representative design data that can be shared among the test partners. Most testbed cases would be functional demonstrations and not involve load and stress testing, so the volume of data need not be large. Early estimates for labor by a user group anticipating such joint testing are below a workyear for each participating site.

Not all possible combinations of CAD and PDM tools need necessarily be on the work list. The business case would put the high value combinations on the early hit list. As the knowledge base grows, the out-of-the-box tool interoperability features should improve and the cost of developing the recipes should decline, making more combinations of lesser value approachable. After a few examples of cooperative method development, and the attending demonstration of business value in adopting projects, the market dynamics will mostly likely change and an alternate funding mechanism will emerge.

XI. Adoption Practices in A&D

A&D development companies are risk averse when considering adoption of tool and process improvements. The dollar value of products made is significant and unproven process activities and IT systems can't prudently put them at risk.

As a result, A&D companies adopt new processes and tools after substantial demonstration of viability and working out of the details. Pilot projects are a common example of such risk reduction. Reference visits to and benchmarking of other organizations who have done it are of similar value. Testbed demonstration activities including published recipes and best practices provide working examples and a starting point for tailoring an implementation. This significantly lowers the cost and risk of an implementation.

The A&D industry would benefit significantly from successful deployment by leading, early adopter companies. From the A&D perspective, a small business jet start up single line of business company might well be sufficient and viable existence proof. Such organizations tend to be agile, have significant time and business pressure to get their product deployed and have few legacy organizational issues to complicate implementation. Few large or mature A&D companies can find a sufficiently unencumbered corner of their organization to undertake such a broad implementation. Furthermore, a small business jet company is arguably in the same business sector but, since they don't directly compete with most A&D companies, would provide a largely open view of the process.

XII. Conclusion

After some years of turmoil, an approach to developing a collaborative design tool environment is at hand. The basis for moving forward lies in the following:

- Current business practices, such as contract structures and the distribution of design authority, mean that read-only access to accurate geometry is sufficient to radically improve the efficiency of collaborative design.
- STEP provides excellent neutral standards for configuration managed exchange of CAD geometry.

- The A&D industry needs validated and published deployment recipes for a few high value heterogeneous PDM and heterogeneous CAD combinations.
- Process mismatch in current A&D organizations must be addressed along with tool implementations.
- Adoption by a single product line business jet company would make an effective demonstration in the eyes of legacy A&D companies.

A&D companies control all the necessary elements here to achieve substantial gains in collaborative design. These are desperately needed in the current business environment. It does require companies to work together to develop recipes and best practices before new high value projects are started.

Appendix A. Detailed Use Cases

1. Scenario: Contributors supply updates to their design data

Participants: System model integration leads, model integration leads from the contributing design organizations

Environment: The Project's home organization has an established PLM system and the Project is building up the complete system design prior to Release. Contributing design organizations have local PLM systems where their design data is maintained. Updates are scheduled regularly. There are identified individuals at the Project and at each responsible design organization that coordinate and execute the update cycle.

Initial conditions: An integrated system model already exists and is to be updated. The design is in progress and precedes "release."

Actions:

As scheduled, the system integration lead calls for an update with general and this-update specific instructions. The instructions might highlight current cycle issues that aim to capture current project design updates or to rectify remaining issues from the prior update.

The integration leads at each organization understand the instructions, coordinate them with their design teams and identify the models to be supplied in response to the call.

In preparation for the update delivery, the organizational design teams and their integration leads prepare the required models, process them through the required quality checks, reconcile issues raised by the check, collect the corrected models and package them for delivery and shipment.

The packaged updates for each contributing organization are transmitted to the system integration leads who accept them, run them through acceptance checks, unpackage them and fold them into the system integrated model database. The system leads in concert with the organizational leads inspect the updated integrated model, identify and reconcile issues, note issues, fixes and responses, and rework the update as required.

The system integration leads certify the completion of the update cycle, publish the issue list with resolutions and note ToDos remaining for the subsequent update cycle.

Concluding state: The system integrated database has incorporated updates and outstanding unresolved issues are documented.

2. Scenario: Contributors supply simplified representations

Participants: System model integration leads, model integration leads from the contributing design organizations

Environment: The Project's home organization has an established PLM system and the Project is building up the complete system design prior to Release. Design organizations have a local PLM system where their design data is maintained. Updates are scheduled regularly. There are identified individuals at Project and at each responsible design organization that coordinate and execute the update cycle.

This scenario illustrates the particular modeling practice of simplified representations at selected locations in the assembly tree. Simplified representations are accurate geometric models that replace or stand in for a large complex assembly. The objective is to provide a more nimble CAD model that is abbreviated in certain respects, but is sufficiently representative for the Project's system level needs. The scenario presumes simplified representations of large assemblies will be supplied in addition to the full model and the system model integrators and users will have the ability to use either and to switch between them as needed for accomplishment of the required functions.

Initial Conditions: An integrated system model already exists and is to be updated. The design is in progress and precedes "release." An update cycle is to be accomplished.

Actions:

In response to a regular call for updates by the system integration leads, the integration leads of the design organization receive the call, comprehend the instructions and identify the assemblies that will need simplified representations. It is most likely that some assemblies already have previously supplied simplified representations and that updates to those will need to be included in this update cycle. As the design evolves and the system integrated model tracks that evolving design content, newly identified assemblies will need simplified representations.

The local organization leads capture this list of updated and new simplified representations and work with their local design teams to see that they are built for inclusion in the delivery.

The simplified representations are run through delivery checks that are tuned for simplified representations. The simplified representations are probably best housed in the CAD assembly tree with the assembly they represent. The simplified representations are packaged along with the other models and submitted.

The system leads unpackage the delivery, find the simplified representations, check them, inspect them for agreement with the call and other issues, reconcile the issues with the delivering organization integration leads, and post the simplified representations to the integrated system model. The resulting issue list with responses and outstanding work is published, including a directory of existing simplified representations so that users of the system model can take advantage of them.

Concluding State: The system integrated database has incorporated updates that include simplified representations and outstanding unresolved issues are documented.

3. Scenario: Design organizations extract design context to support their work

Participants: Contributing organizations' model integration leads and design team leads, system model integration leads.

Environment: The Project's home organization has an established PLM system and the Project is building up the complete system design prior to Release. Design organizations have a local PLM system where their design data is maintained. Updates are scheduled regularly. There are identified individuals at Project and at each responsible design organization that coordinate and execute the update cycle.

The design organizations utilize design data from other organizations in the design context for their assigned work. In this scenario, that data from others' work is drawn from the system integrated model database. At selected times, potentially asynchronous to the system model update cycle, all or subsets of the system integrated model are distributed back to the contributing organizations. These times would be at stable, semi-coherent/consistent, quiescent periods for the system database, such as just after an update cycle. It might be that, after a few update iterations, this could be executed as a closing phase in the update cycle.

This distribution would have to be consistent with the project security model. It will likely be that some contributing organizations will not be allowed to access or possess portions of the project system model integrated database.

Since this serves the needs of the participating development organization, the workflow should avoid involvement of the system integration leads. It would probably lead to direct access by the local organization integration leads to the system integrated model so that they can serve themselves.

Initial Conditions: An integrated system model already exists and is to be updated. The design is in progress and precedes "release." An update cycle is to be accomplished.

Actions:

The local integration leads develop a list of system integrated database contents to extract. Consideration needs to be given to meeting security-based access rules, complex issues surrounding bringing-back-home your-own-previously-submitted models, extracting common library parts and extracting simplified representations vs full models.

The models are extracted, packaged and transmitted to the local organization.

The local integration leads unpackage and place the models in their local PLM repositories. Consideration needs to be given to read/write state (surely it is read-only), version control (align with existing tree, version number tracking), setting PLM metadata and CAD parameters, incorporation of new models (find a place in the tree, deal with conflicts, find a recipient on the local design team). Identify issues, reconcile issues with system integration leads and local design team leads, feed back relevant issues to system integration leads, publish the update for use by the local design teams.

Concluding State: The design database at the design organizations has been updated with models from the system database. The conditions, issues and assumptions of the update are documented. The updated model and its issue list are available for use by the design team. The models representing others' work are read-only and security/access rules have been applied.

4. Scenario: Non-CAD workers use the integrated model

Participants: Project participants at the system level, at contributing design organizations and at other locations, that are not CAD designers.

Environment: The Project's home organization has an established PLM system and the Project is building up the complete system design prior to Release. Design organizations have a local PLM system where their design data is maintained. Updates are scheduled regularly. There are identified individuals at Project and at each responsible design organization that coordinate use of the system model.

The business situation supports a community of users who are not CAD users and not CAD authors. It is presumed that they will not use a CAD tool but will use a visualization tool. The visualization models would appear to follow the same organization (assembly tree, BOM, structure) and naming/number scheme as the CAD models.

For now this scenario assumes only the system integrated database would be accessed. That is, this activity does not draw upon a drill down to the local PLMs. If simplified representations are part of the system integrated model, the visualization users might default to visualizations of the simplified representations. If the system integrated model also contains the full CAD assembly as well as the simplified representations, the visualization users might switch to the visualization of the full CAD assembly and drill down into that.

Access rules are in place, analogous to access/security rules for the CAD models.

Initial Conditions: The project integrated system master model database has had an update recently and is in a stable state. One result of the update cycle is that visualization datasets for the updated model are available in the database. Users have accounts, installed visualization client software and have been trained.

Actions:

Some use cases depend upon the implementation. It might be that visualizations to support a meeting or intended use have to be prepared ahead of time. It might be that live access to the visualizations in the system integrated database can only be supported in some IT-meaning of "local" due to network or server response times. It might be that the implementation provides only read access for viewing or it might be that mark up, annotation and capture is supported by the implementation.

For all uses, configuration management must insure that the visualizations are consistent with the system integrated model CAD contents, at least at relevant epochs. For example, the visualizations might be as of the last update cycle and therefore not track interim changes (if any). Users of the visualizations must be able to determine this date.

Participants access the system integrated model database that would serve them with lightweight visualization models rather than the full native CAD model.

These uses presume the visualizations served are not saved locally or outside the project PLM system. There is no requirement in this scenario to preserve and configuration manage these served visualizations. If mark-ups are made in the activity, they are to be captured, associated with either the activity or the design and configuration managed.

Concluding State: Participants can successfully access and utilize elements of the integrated model and issues that are discovered are documented and captured.

5. Scenario: Library parts are managed in a design model update cycle

Participants: System model integration leads, model integration leads from the contributing design organizations

Environment: The Project's home organization has an established PLM system and the Project is building up the complete system design prior to Release. Design organizations have a local PLM system where their design data is maintained. Updates are scheduled regularly. There are identified individuals at Project and at each responsible design organization that coordinate and execute the update cycle.

The library of standard and common commercial parts has been established. For now, assume that the project has already established the library architecture, standards and conventions and that each organization has a local copy of the library. These are named consistently in the pre-arranged fashion. The geometry content for all copies of the library need to be consistent but don't need to be identical. For example, the copy used at the design organization might have fine detail, internals or even a complete sub-assembly tree, eg for a connector. The system database part might have the same identifier, but may have a simpler geometry with fewer fine details and maybe even represent the subassembly as a single part. Then, when the design organization provides an update, the part number is in the assembly tree/BOM but the geometry model is not shipped, shipped & not used, or shipped and replaced upon integration into system database.

Initial Conditions: A common library exists and each design organization is using it. An integrated system model already exists and is to be updated. The design is in progress and precedes "release." An update cycle is to be accomplished.

Actions:

In response to a regular call for updates by the system model integration leads, the integration leads of the design organization receive the call, comprehend the instructions and identify the assemblies that will be updated. It is most likely that some assemblies already have previously supplied which contain library parts and that the evolving designs use additional library parts or change usage in existing designs.

As part of preparing their submission, the local integration leads check the models using checks tailored to library parts. Unqualified library parts are to be identified. The integration leads might work these issues with the design teams directly if time permits and resolution is a simple name correction or part selection. If rework is not practical, the issue will be logged and the offending part rejected from the submission, potentially leaving a gap.

The local library part file in the submission might be replaced, at the sending organization or at the receiving organization, with an equivalent but different model. This replacement might be more like a simplified representation that reduces the assembly tree or omits internal detail.

There might be some issues that are resolved, perhaps temporarily, by applying renaming rules to the offending library part and including it in the submission. These would be documented with the submission. A common disambiguating rule is to prepend the CAGE code (of the submitting design organization OR the COTS vendor). The system integration leads would note and track these parts so that subsequent rework and clean up could be accomplished.

Concluding State: The system integrated database has incorporated updates that include library parts and outstanding unresolved issues are documented.

Acknowledgments

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

The author acknowledges the interoperability use cases from unpublished work by the Aerospace and Defense Industry Council.