Impact of System-Level Engineering Approaches on the Aerospace Development Cycle

Fred Emch
Technical Group Leader
EDS
(314) 264-8430
fred.emch@eds.com

Digital Transformation of the Aerospace Product Lifecycle

Aircraft Development Process Flow Outline

Phase 1

Requirements & System capture from Aircraft source document

Phase 2

Changes to source requirements

Phase 3

Propagation of changes and update of product structure

Process Flow – Phase 1

Requirements & System capture from Aircraft source document

- Define high-level aircraft requirements using the System Architecting tool called SLATE
- Derive lower-level requirements and link back to original source requirements
- Create decomposition representing the physical aircraft system
- Link derived requirements to decomposition
- Link requirements to Excel spreadsheet

What is SLATE…

- SLATE (System Level Automation Tool for Engineers)
- A standalone product that provides the means for taking requirements, systems engineering, and product knowledge to the virtual enterprise
- Computer-Aided Systems Design Groupware that
  - Captures systems views
  - Captures & relates requirements into those views
- So requirements (aka the customer) can influence daily design decisions towards compliance

What does SLATE do?

- Helps organizations capture, relate, and manage information about a company's products and customer requirements
- Enables continuous design alignment with customer concerns across the virtual enterprise
- Capabilities have been applied to many aspects of product development including…
  - Requirements definition
  - Product design
  - Test management
- This results in built-in compliance—this produces products customers want, delivered ahead of schedule

Some of SLATE's user extensible building blocks...

Folders for organizing information—Requirements, issues, etc..

Notes provide a project notebook for documenting decisions, action items, critical issues, test results, etc.

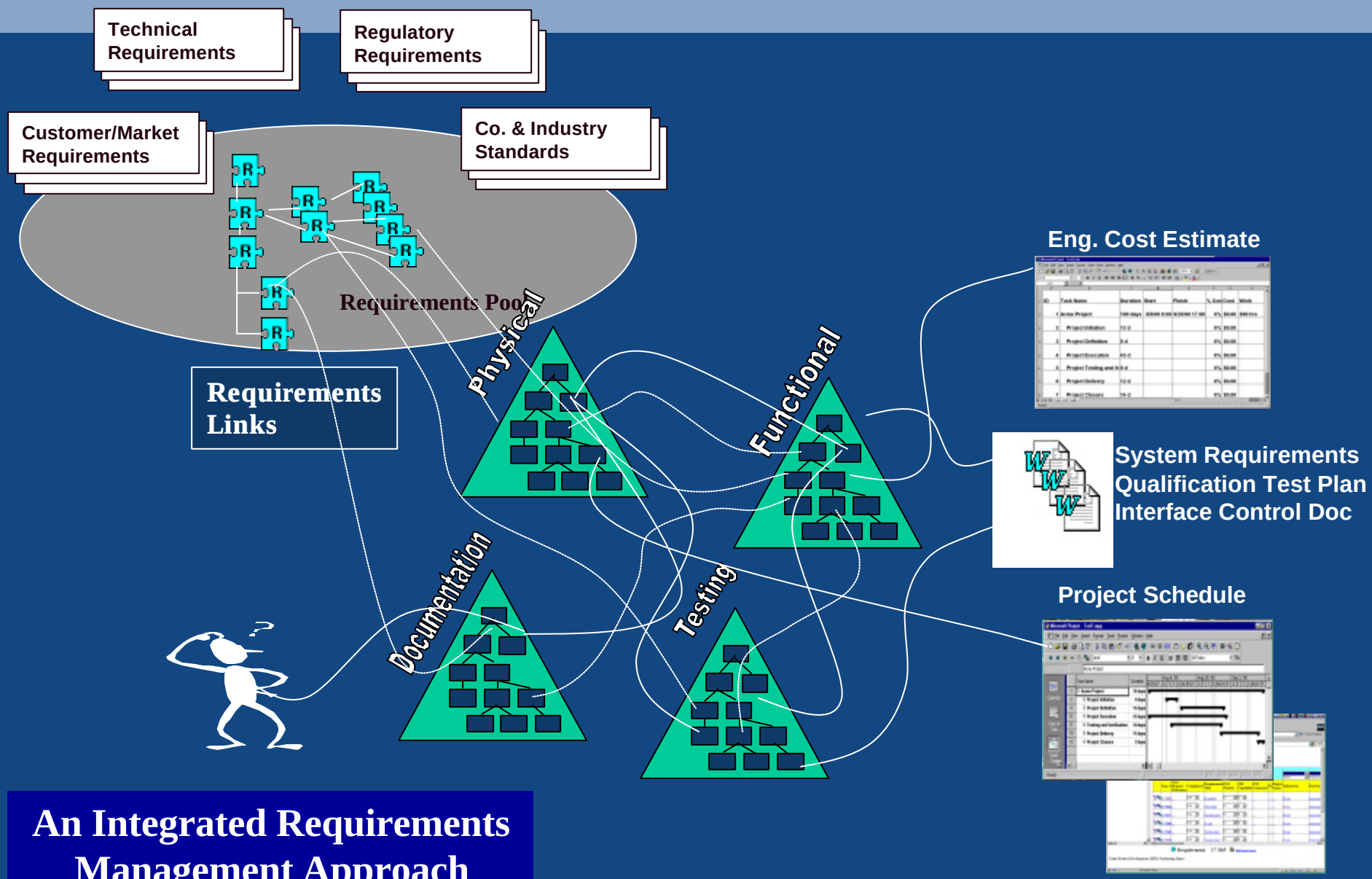
Requirements capture the intent of the product design, goals, performance objectives--what you are going to verify & how you know you are done.

Building Blocks (what we call Abstract Blocks--AB's) are used to build things--products, organizations, test systems, processes, etc.

Documents provide a customer input/output media--such as test documents

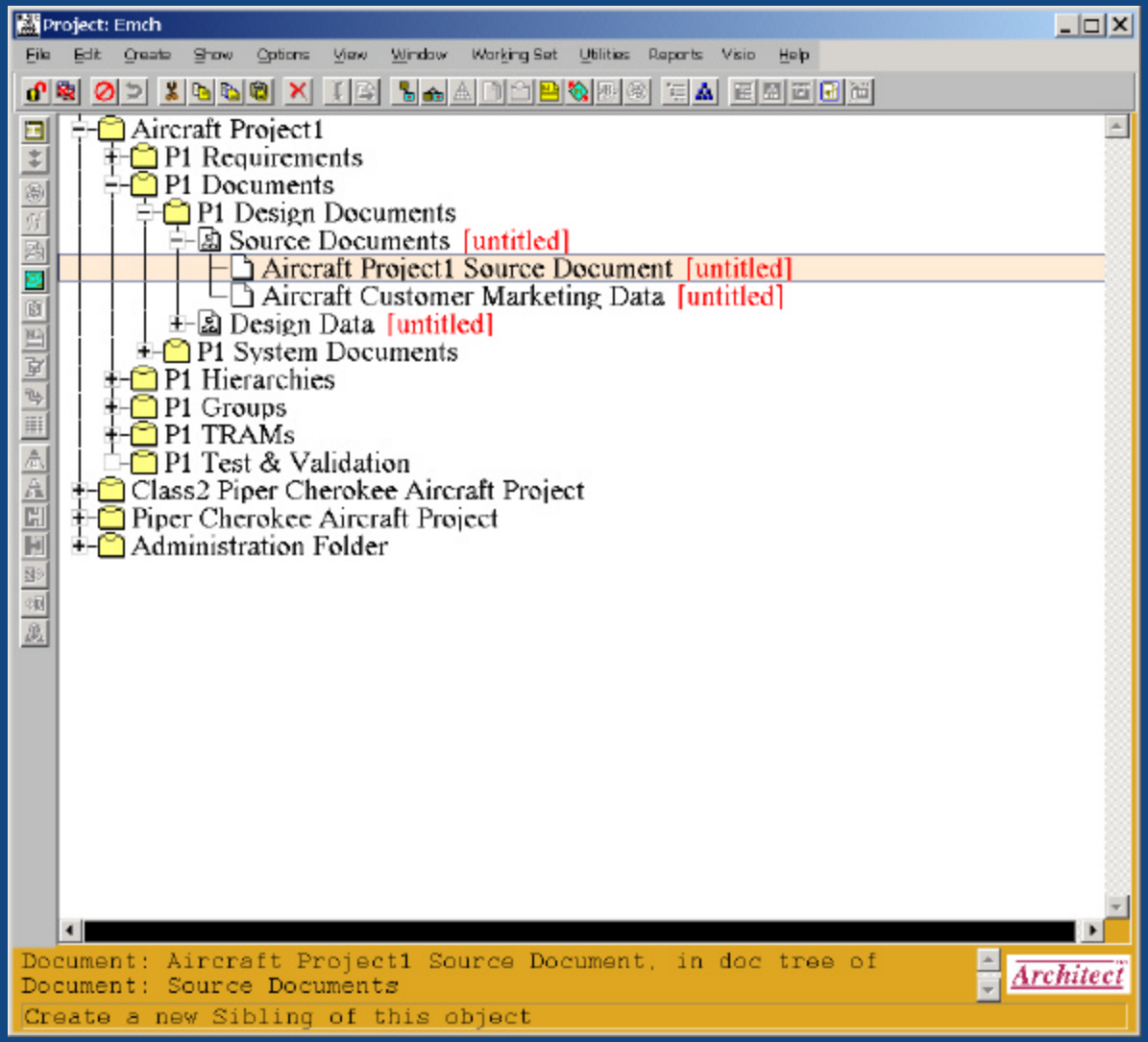
Budgets relate measurable goals to product designs (schedule, cost, weight, etc...)

SLATE Overview



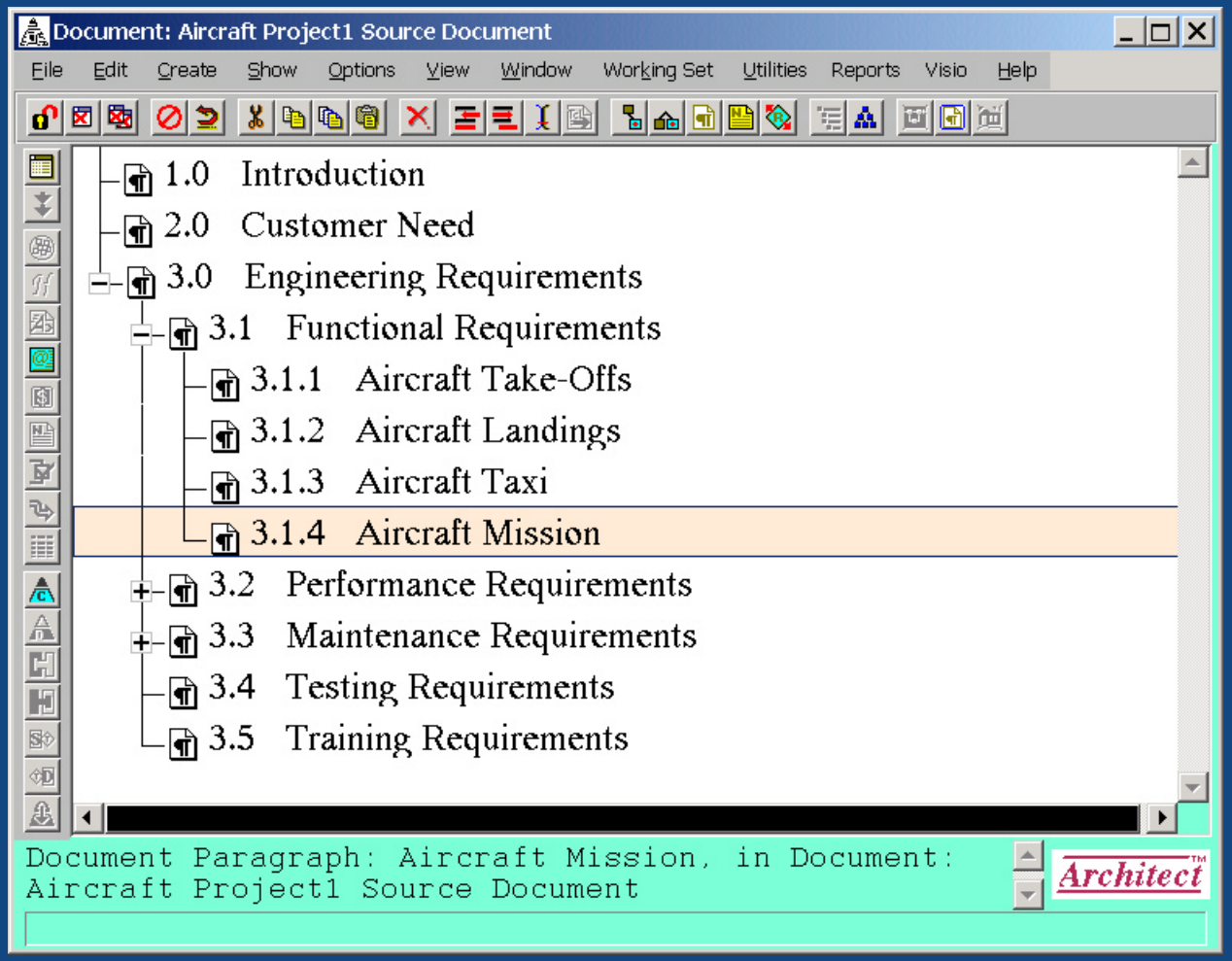
An Integrated Requirements Management Approach

Process Flow – Phase 1



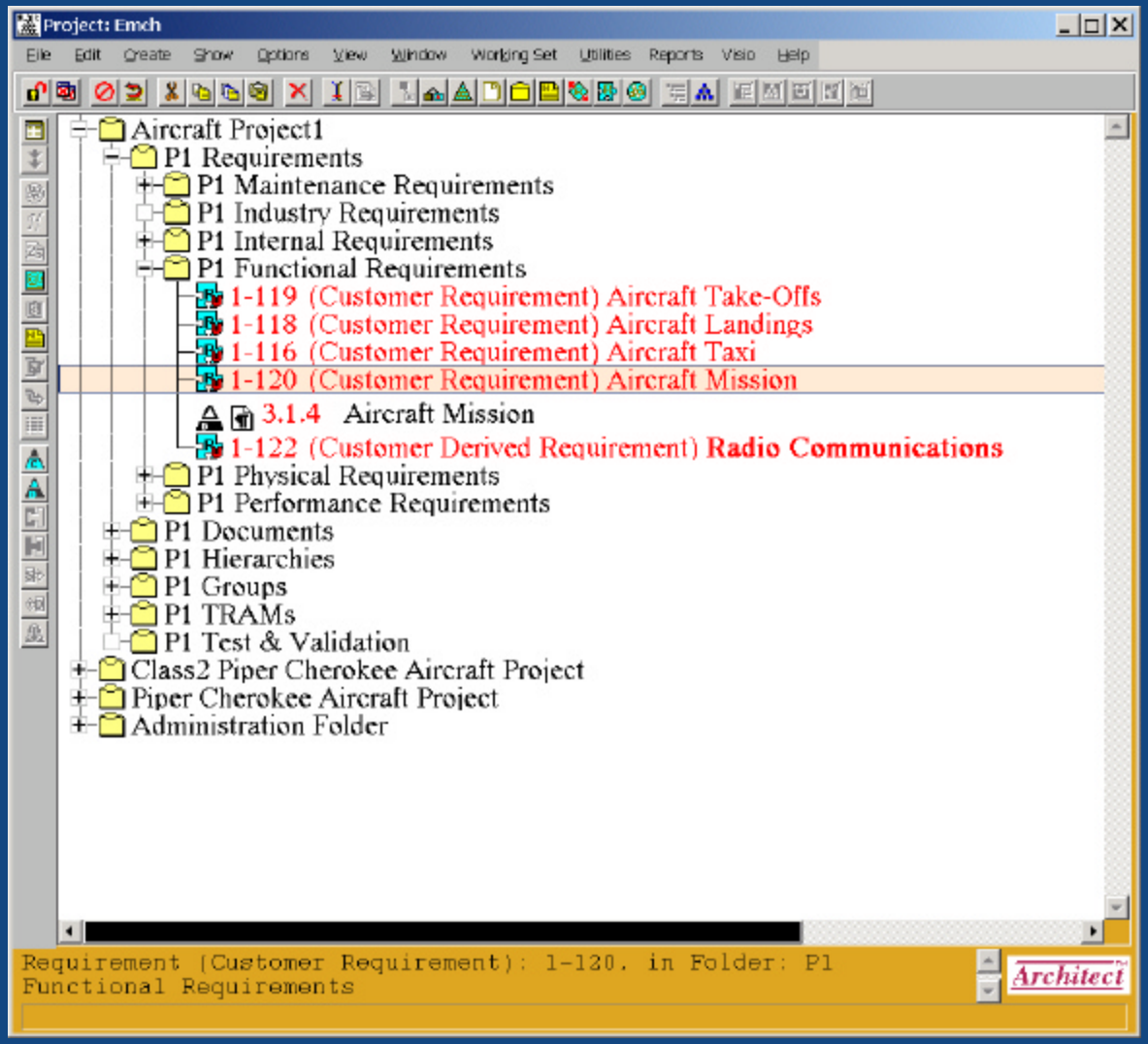
- Source document is imported into SLATE

Process Flow – Phase 1



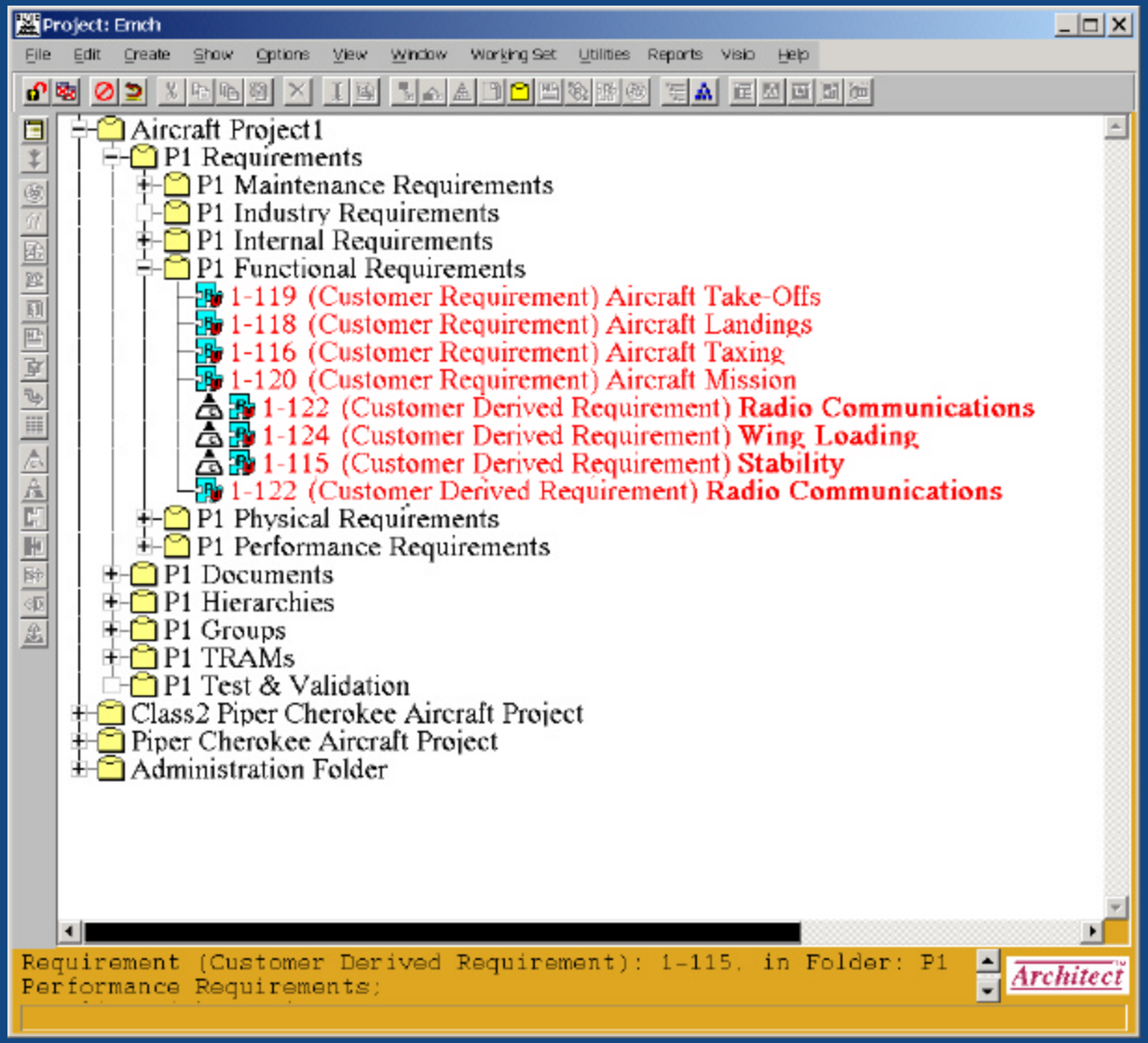
- Source Document in SLATE
- Paragraphs in source document become paragraph Objects in SLATE

Process Flow – Phase 1



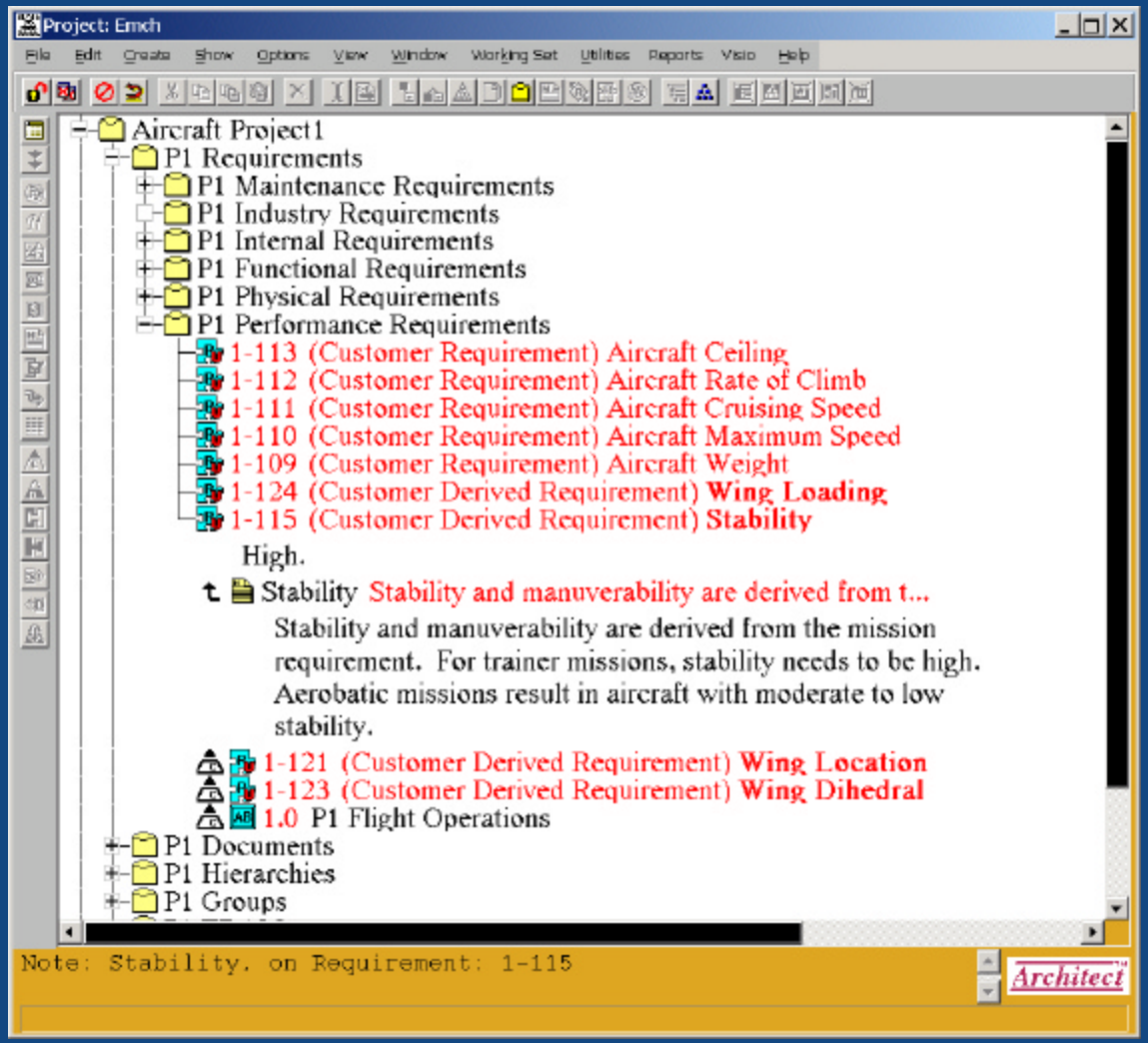
- Requirement objects are generated from, and linked back to, source document paragraph objects
- Source paragraph becomes a “Defining Object” (D)
- If changes are made to source document, requirements get updated

Process Flow – Phase 1



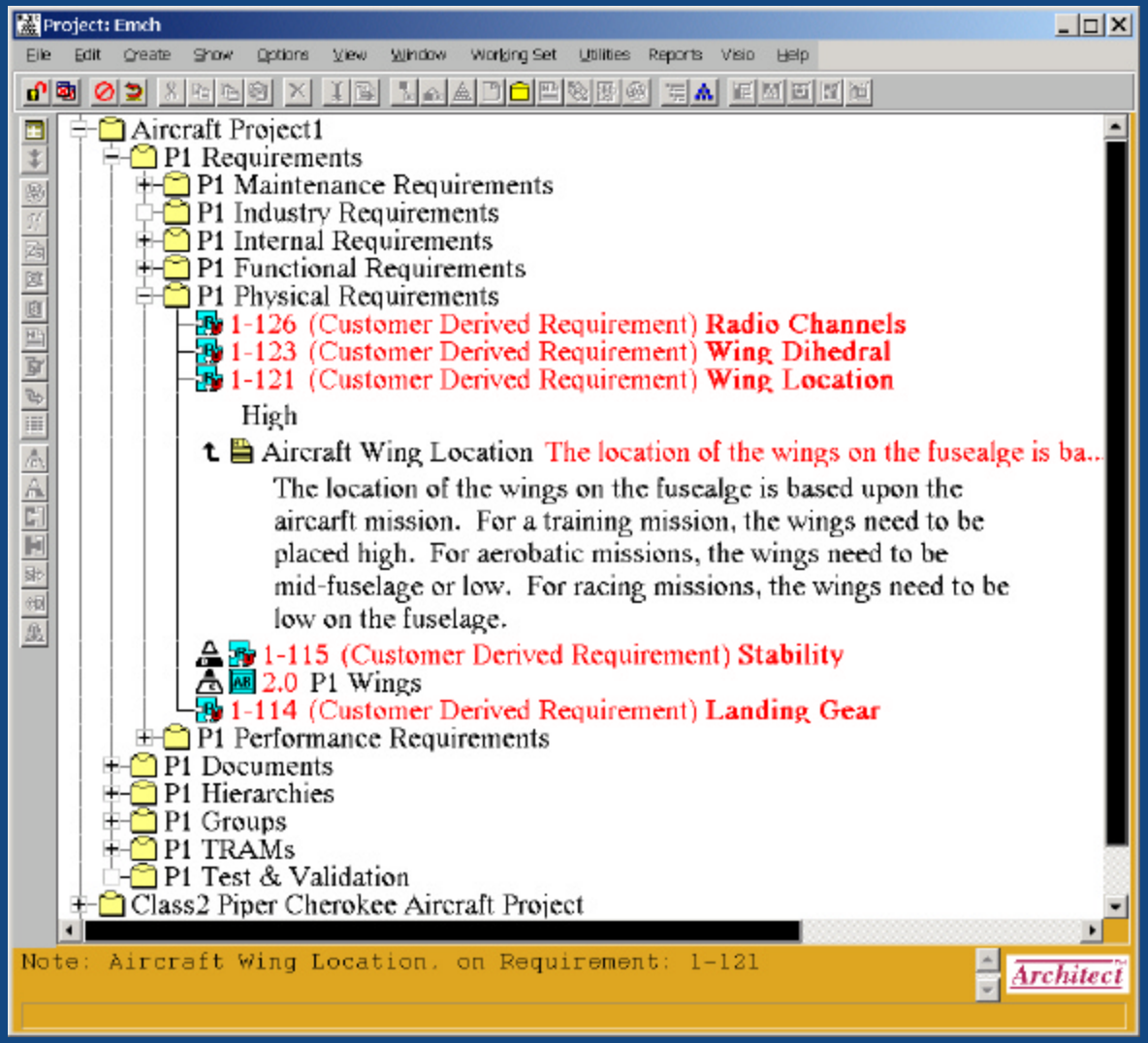
- Customer requirement has derived requirements as “Complying Objects” (C)

Process Flow – Phase 1



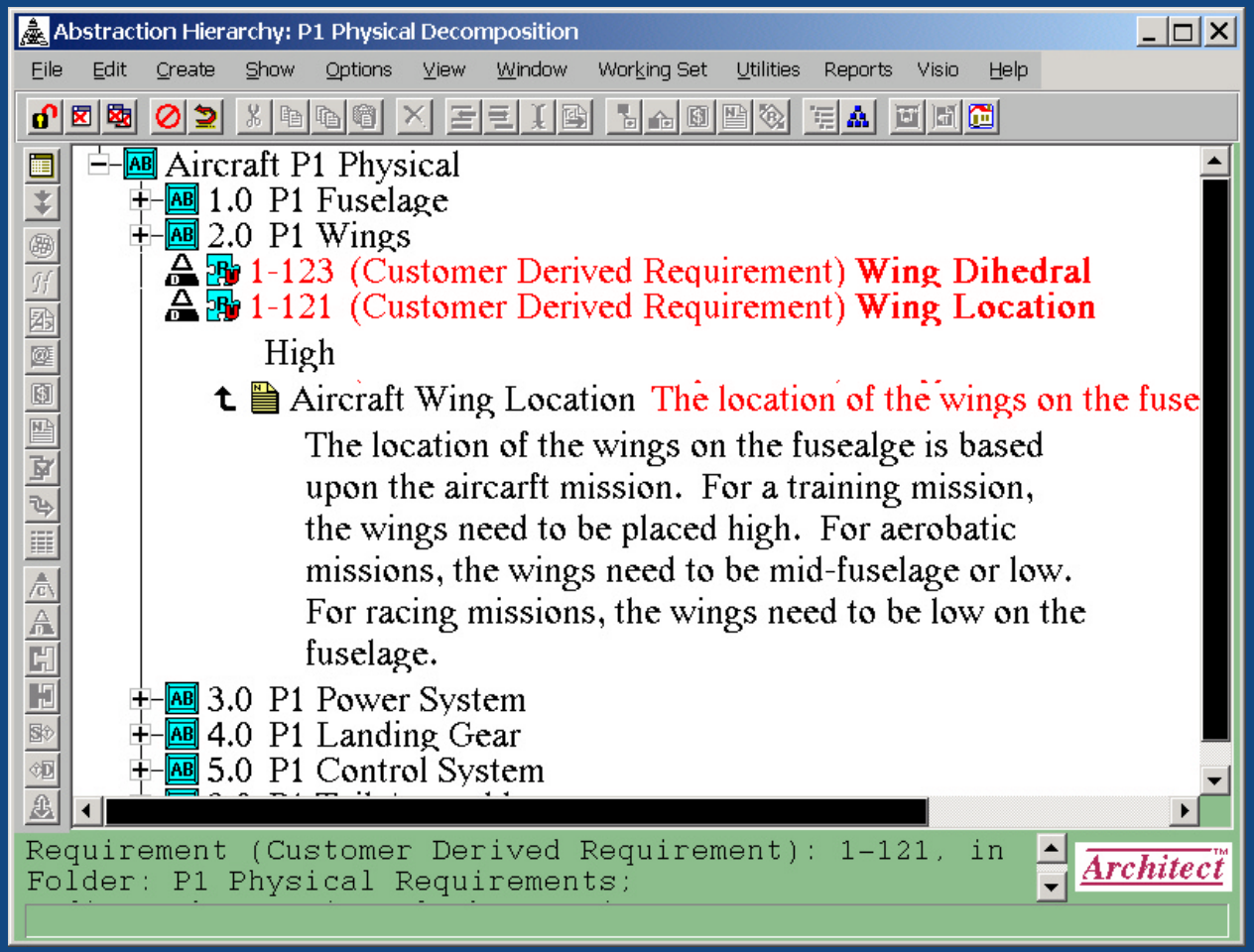
- Query for complying objects at each level of requirement hierarchy
- Attached Note Objects record design decision-making process

Process Flow – Phase 1



- At each requirement level, applicable Defining Objects can also be queried
- Complying objects include nodes of the physical decomposition

Process Flow – Phase 1



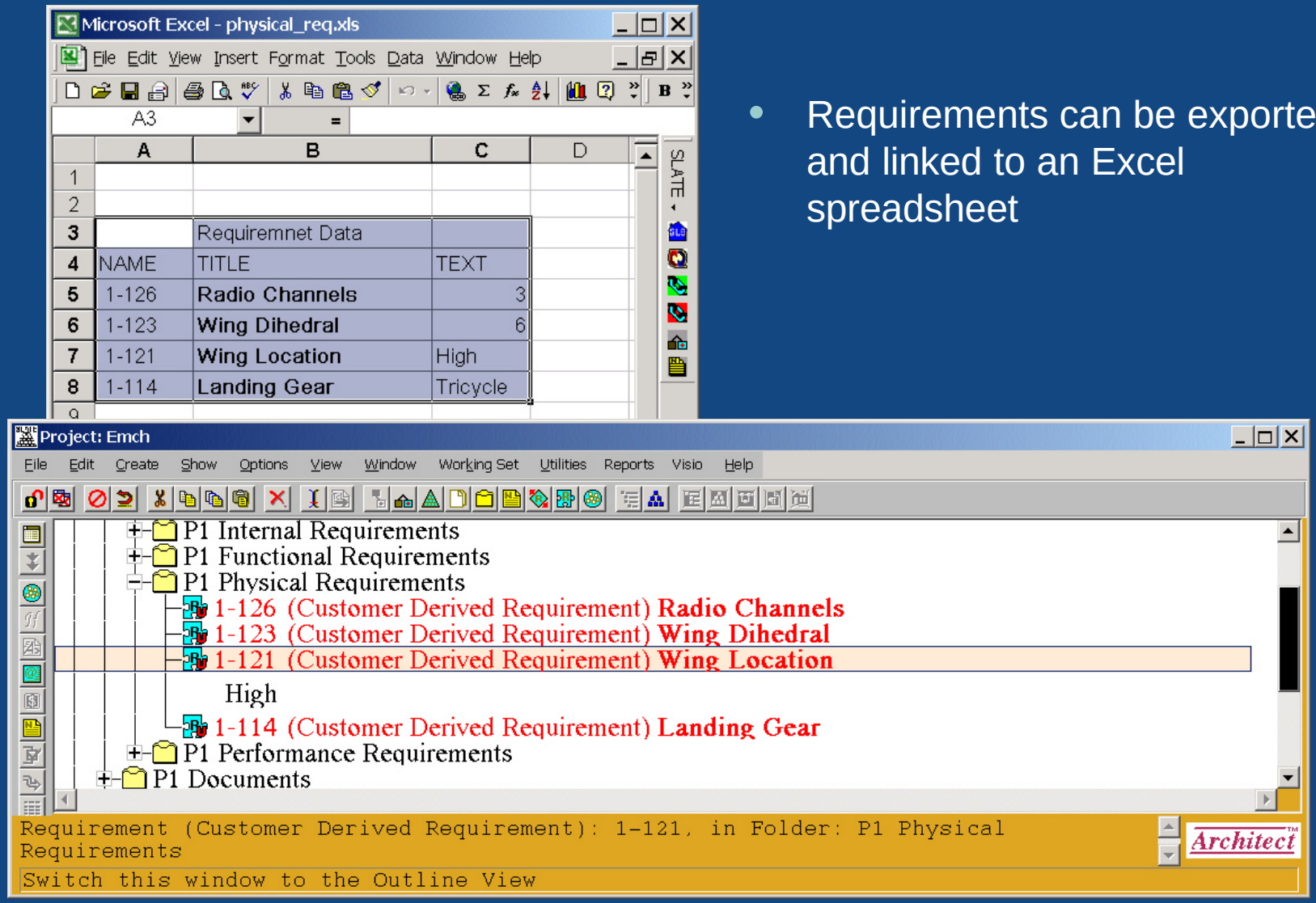
- System decompositions can be physical, functional, test, materials, maintenance, training, etc.
- Applicable requirements are linked to appropriate nodes of multiple decompositions
- Establishes built-in compliance with requirements to product structure

Process Flow – Phase 2

Changes to source requirements

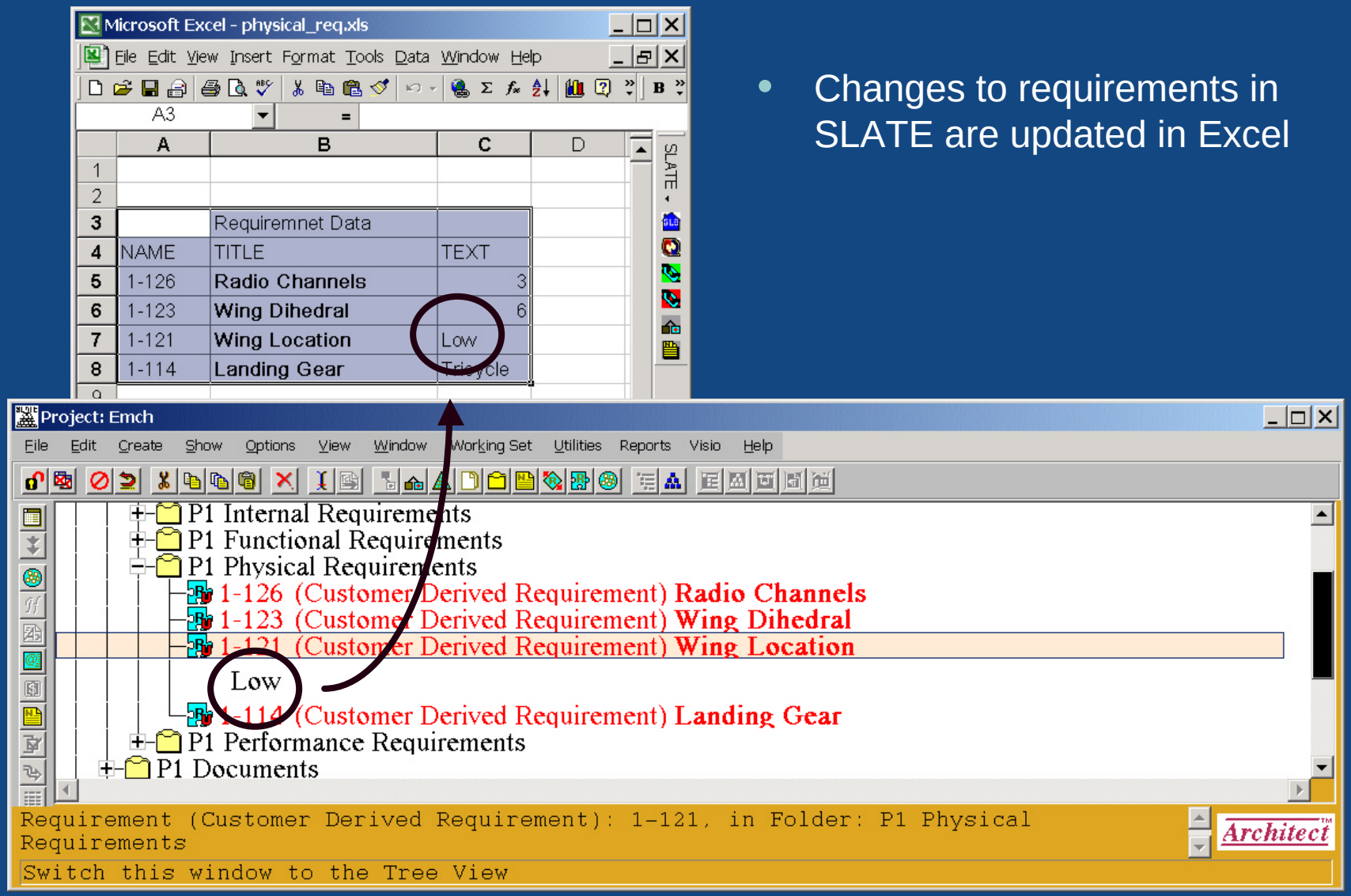
- Change Mission from Trainer to Aerobatic
- Follow change process in SLATE with requirements and AB decomposition
- Cascade changes through requirements tree resulting in new wing location
- Requirement change updates Excel spreadsheet

Process Flow – Phase 2



- Requirements can be exported and linked to an Excel spreadsheet

Process Flow – Phase 2



- Changes to requirements in SLATE are updated in Excel

Process Flow – Phase 3

Propagation of changes to the product structure

- Update to Excel triggers an update to UG/Knowledge Fusion (KF)
- KF, in turn, triggers update to wing location in CAD model
- KF also updates other systems such as landing gear, horizontal location, etc.
- Creates fully editable parametric geometry in Unigraphics

Aircraft Design

the traditional way

Two separate worlds !

Requirements

Process

1. Capture requirements and rules
2. Evaluate against new criteria
3. Create concepts

Tools

SLATE, Spreadsheets,
In-House Code, KBE

Geometry

Process

1. Create concept geometry
2. Add “rules” (constraints)
3. Evaluate and modify

Tools

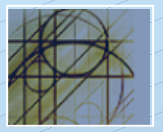
Unigraphics, In-House Scripting Routines

Aircraft Design Using Knowledge Fusion

technology breakthrough

A unique capability that blends the power of Requirements and Rules Capture with the utility of high-end CAD

Requirements



Geometry

Process

1. Capture requirements, rules
2. Evaluate against new criteria
3. Create Concepts

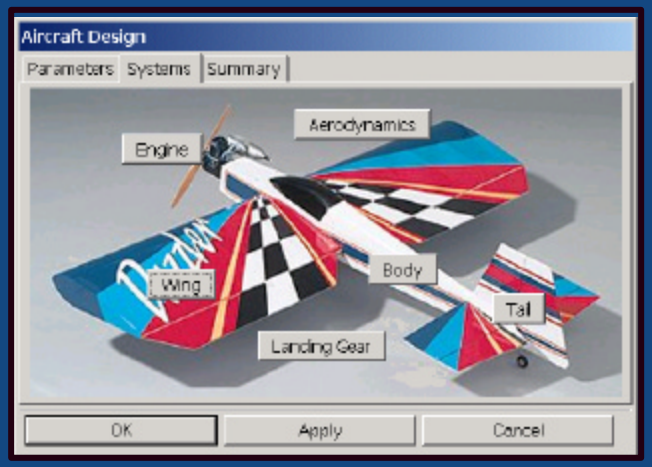
1. Create concept geometry
2. Add “rules” (constraints)
3. Evaluate and modify

Tools

Unigraphics

Knowledge Fusion

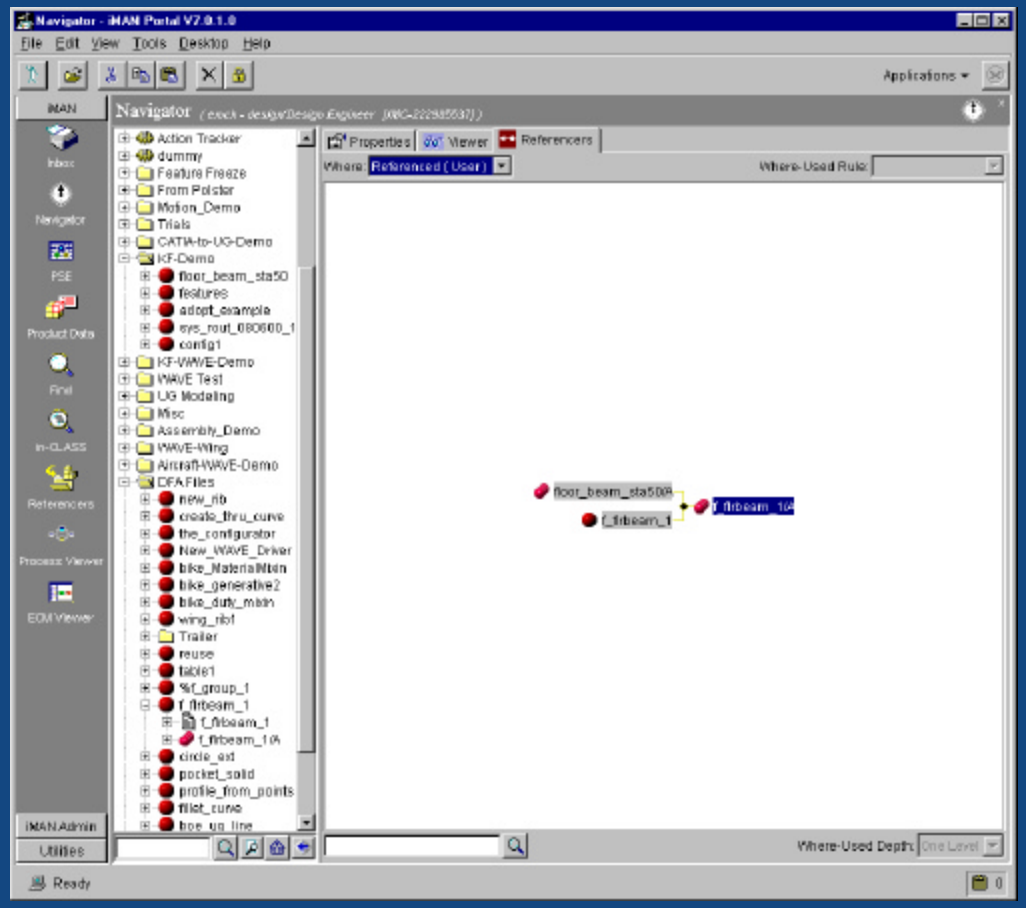
Knowledge Fusion Aircraft Creator



- Captures aircraft design processes
- Creates CAD geometry from input rules
- Enables editing which results in gross topology changes
- Optimization and analysis
- Links to external data sources (Access, Excel)

KBE/PDM Integration

- Knowledge Fusion programs are stored in the Teamcenter database in a secure library
- Searches can be performed with Keywords to locate a program
- The Teamcenter database can be queried to find out which UG part files would be affected with a change to a program



TOP WORK

TOP WORK

Thank You

Impact of System-Level Engineering Approaches on the Aerospace Development Cycle

Fred Emch
Technical Group Leader
EDS
(314) 264-8430
fred.emch@eds.com