IBM Innovate 2011

A Linked Data Approach to Integrating Software, Electronics, and Mechanical Domains

Thomas Konrad
IBM Rational, Systems Business Unit Executive Asia Pacific
thomas@sg.ibm.com

IBM Software
Innovate2011
The Premier Event for Software and Systems Innovation

Software. Everyware.

Please Note:

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Agenda

- **The Systems Engineering Challenge – Rational Perspective**
- Engineering Lifecycle Management needs
- Tool integration patterns for Engineering
- Summary

Challenges in Complex Products Development

Collaboration between domains is still manual..

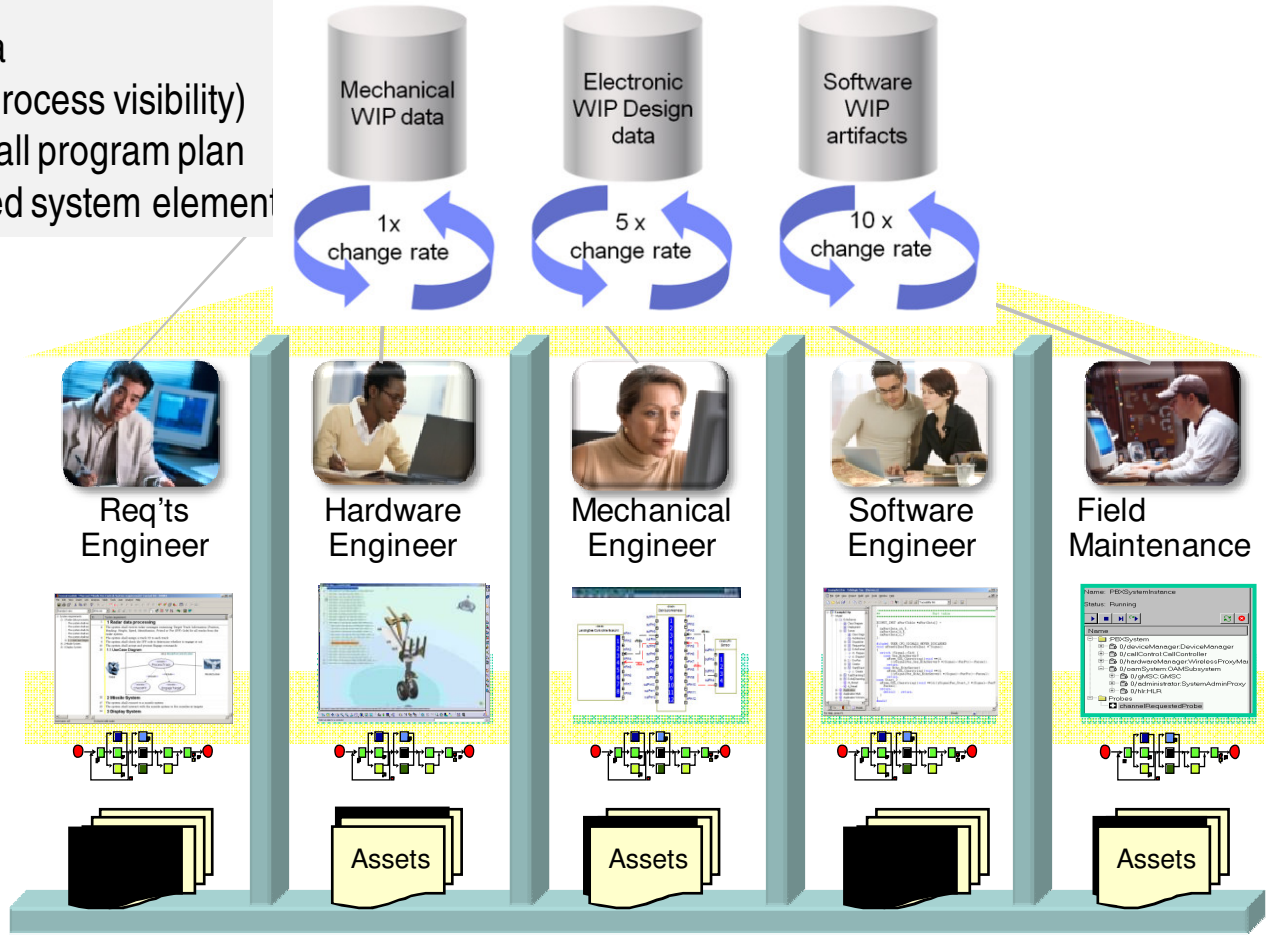
- ✓ Related and dependent data
- ✓ Progress on related tasks (process visibility)
- ✓ Progress related to the overall program plan
- ✓ Visibility to changes in related system element

Many Different Roles

Best of Breed Applications, Domain Specific Processes

Domain Specific Program Assets

Program Master Plan



Business results of increased product complexity driving critical imperatives for product development and delivery

Business View

Product missed customer needs	46%
Late to market/missed demand	33%
Poor commercialization / promotion	26%
Product quality	24%
Pricing	23%
No clear product differentiation	19%

The CIO's Guide to the PERFECT Launch: Translating Innovation to Business Benefit, AMR Research, 2005

Engineering Opportunity

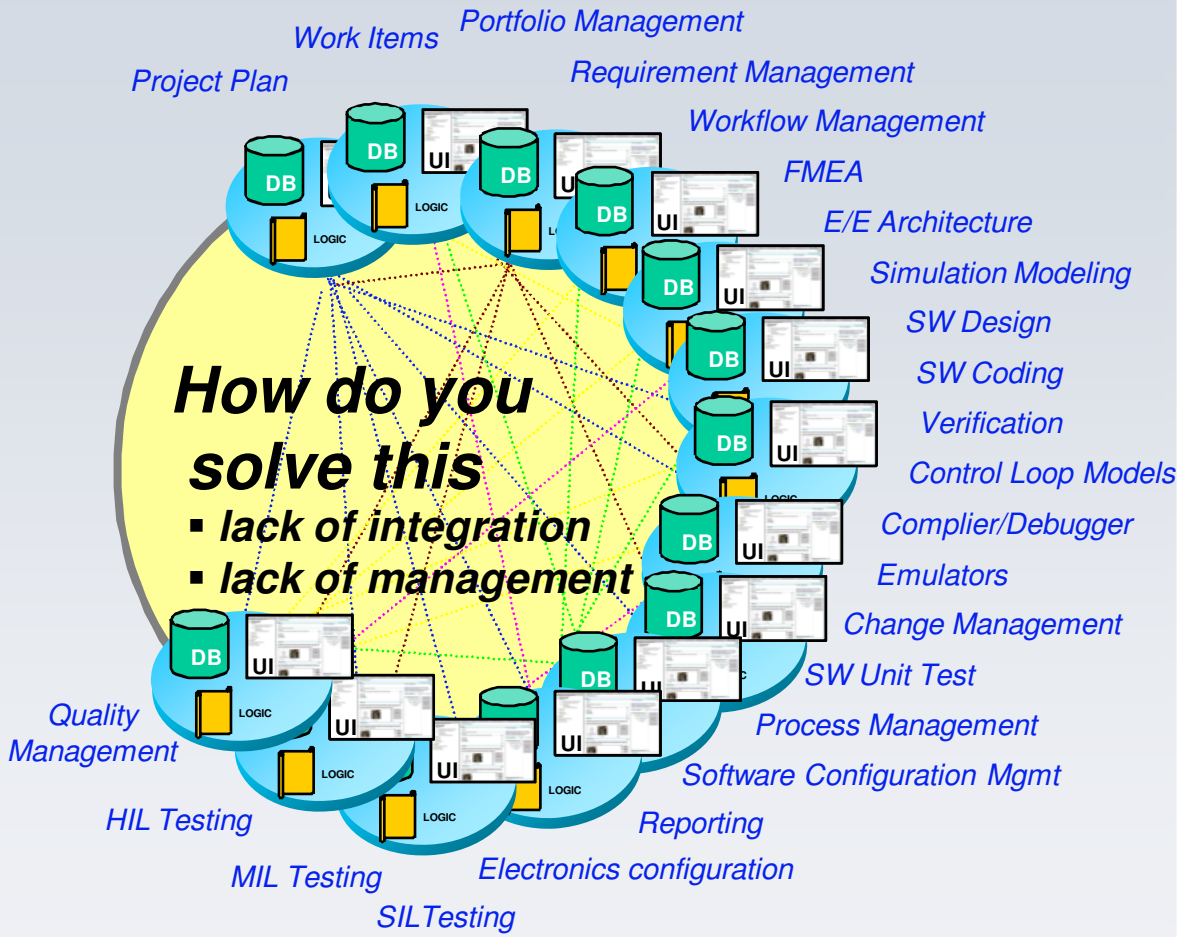
Improve communication and collaboration across disciplines	**71%**
Increase visibility into status of requirements	49%
Increase ability to predict system behavior prior to testing	46%
Implement or alter new product development processes for a multi-disciplinary approach	43%
Increase real time visibility of product Bill of Materials (BOM) throughout the development process	39%

Aberdeen Group, System Design: New Product Development for Mechatronics, Michelle Boucher, David Houlihan, January, 2008

5

Engineering environments are highly fragmented -
the challenge to connect them is increasing exponentially

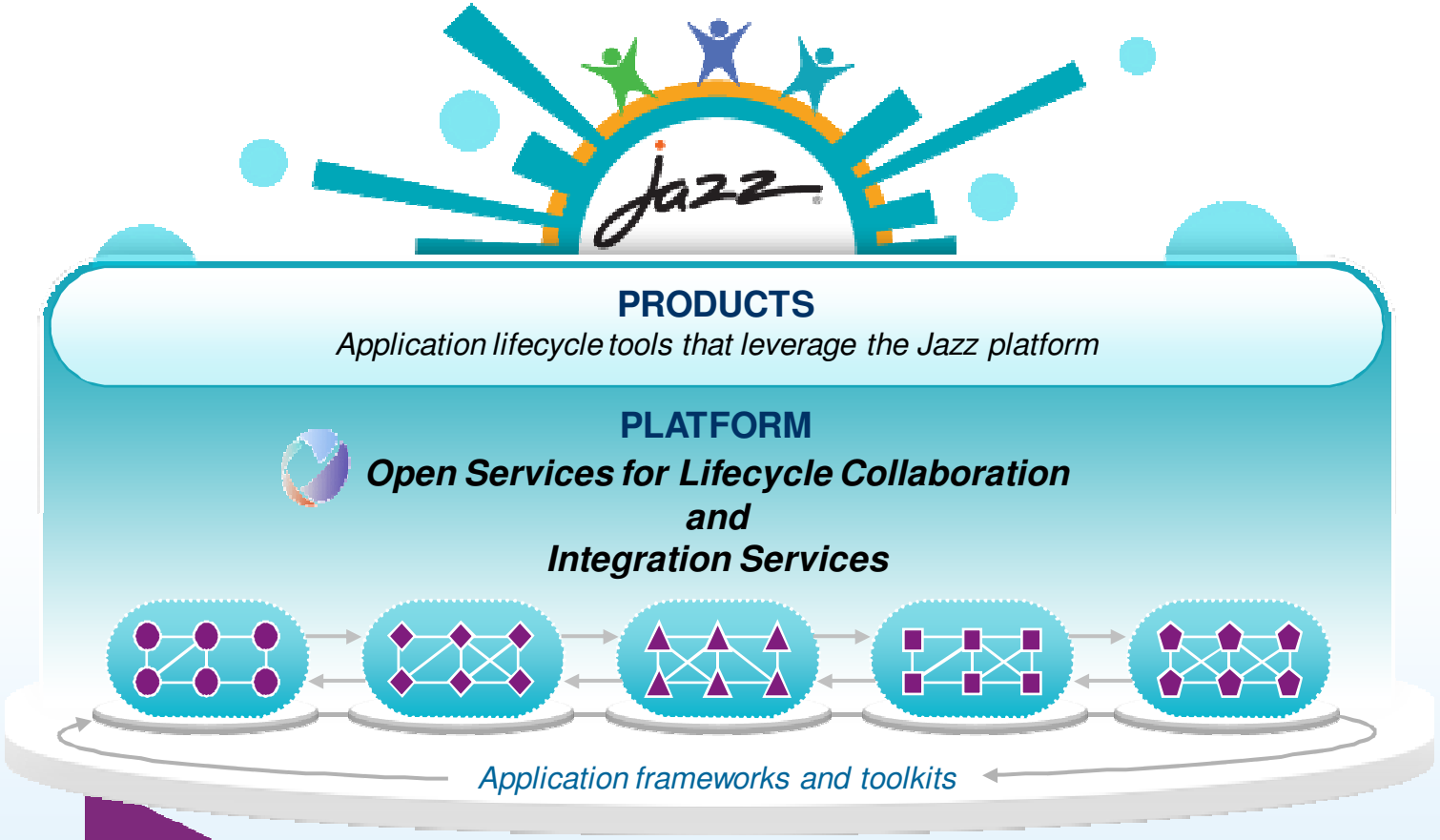
- **Traditionally, each tool came with its own**
  - **UI** - Web and desktop presentations of views and tasks
  - **Logic** – Workflow, process, search, query, scale, security and collaboration
  - **Storage** – individual files on workstation or servers: how to ensure availability and traceability?
- **Resulting in...**
  - Brittle/poor integrations
  - Silos everywhere
  - High cost to maintain and administer
  - Low re-use



6

OSLC and the Jazz Platform

IBM's Jazz platform leverages OSLC, providing open collaboration across the software and systems lifecycle



OSLC: simplifying lifecycle integration

Cross lifecycle scenarios implemented to open specifications

426 **registered community members from** ***127*** **organizations and growing**

Open Services for Lifecycle Collaboration

Open interfaces. Open possibilities.

Increasing Adoption

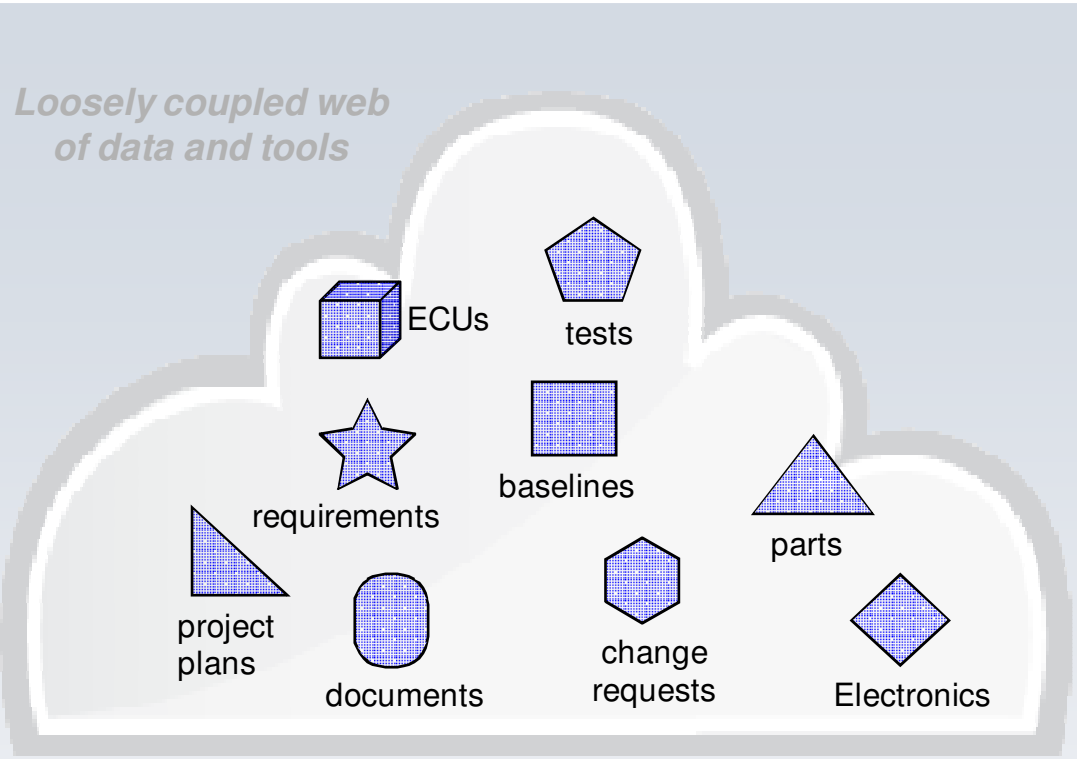
- New Eclipse OSLC SDK project proposal
- Open source community projects
- Clients integrating in-house tools
- IBM business partners releasing integrations
- Growing number of published specifications

"With OSLC's open and scenario-based approach, ***businesses benefit*** *from the ability to* ***tie disparate tools together****. This collaborative approach gives our consultants the* ***flexibility*** *to make lifecycle tool choices based on* ***specific client project demands****."*

Randy Vogel, Accenture

[1] Open-services.net

Jazz Integration Architecture enables a loosely coupled "web" of engineering data



- Federated integration architecture
- Provides common, cross-product capabilities (search, query, report, process, etc.)
- Integrate tools multi-vendor and in-house tools
- No duplication/synchronization of data
- Incrementally add tools and capabilities
- Leverage existing tool investments and best of breed capabilities
- Upgrade parts individually

9

Goals

- Practical
  - Improve user experience of team members trying to discover, understand and reuse engineering artifacts
  - Bridge semantic gap between domain and tool terminology
  - Improve collaboration and communication across disciplines
  - Reduce redundant information entry/copying
  - Reduce integration cost and complexity
  - Improve and automate processes (find the “gaps”)
- Aspirational
  - Advance key architectural tool integration patterns
    - Configuration management and versioning
    - Product line engineering and variability
    - Multi-model integration
  - Foundation for analytics and discovery
    - Watson for engineering?

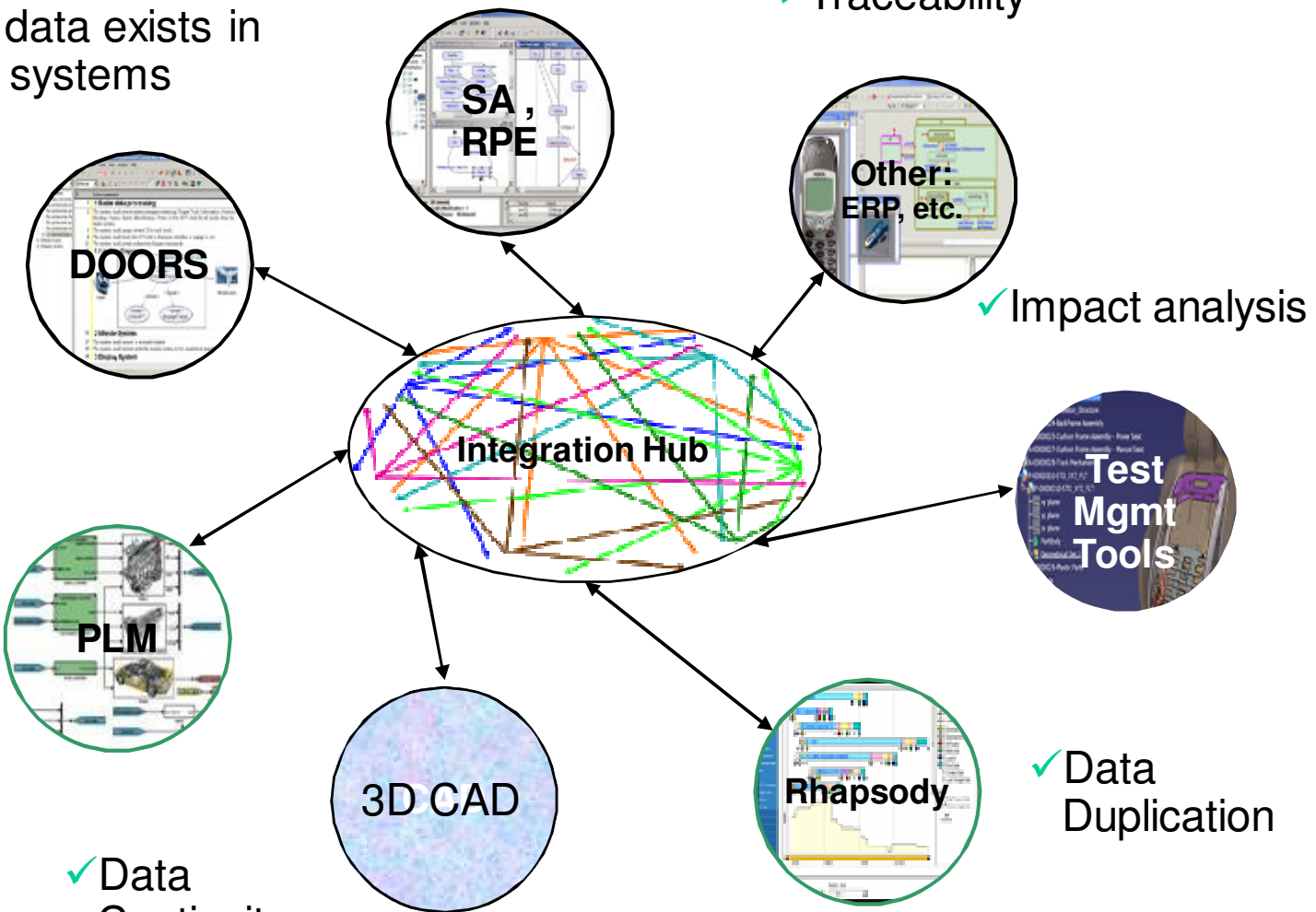
Agenda

- The Systems Engineering Challenge – Rational Perspective
- **Engineering Lifecycle Management needs**
- Tool integration patterns for Engineering
- Summary

The Solution Approach

✓Product data exists in different systems

✓Traceability



✓Impact analysis

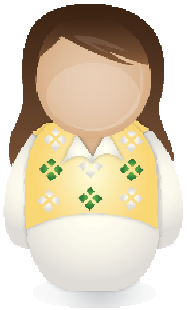
✓No peer to peer integration

✓Data Continuity

✓Data Duplication

Each tool domain has its own organizational structure

How do I find out who is dependent on this interface?

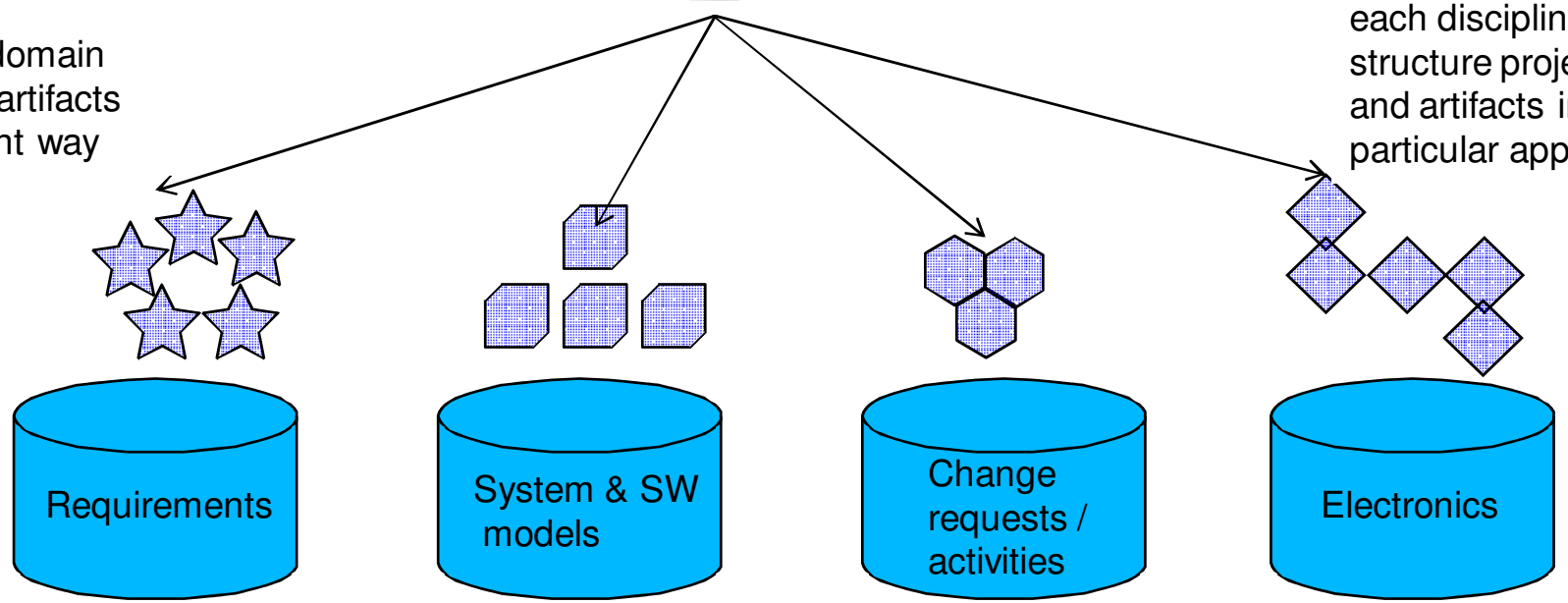


Which artifacts are used in Product X?

Who is working on the product use case 'foo'?

each tool domain organizes artifacts in a different way

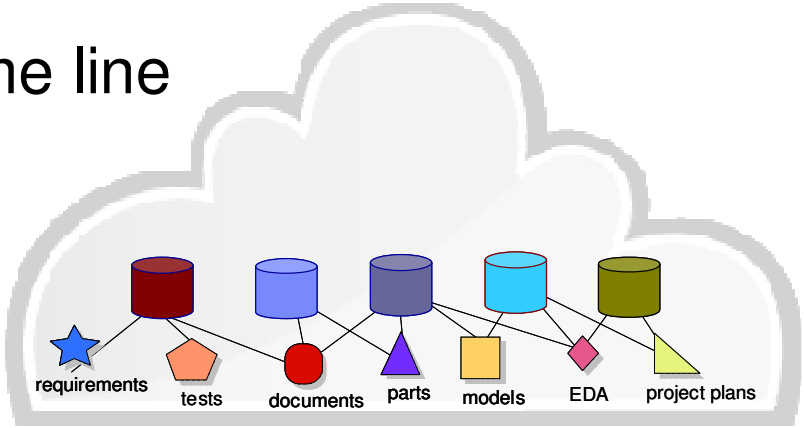
each discipline may structure projects and artifacts in their particular approach



No shared view of project/product configuration
No single point of access or integrated view/perspective

Integration Architecture Challenges

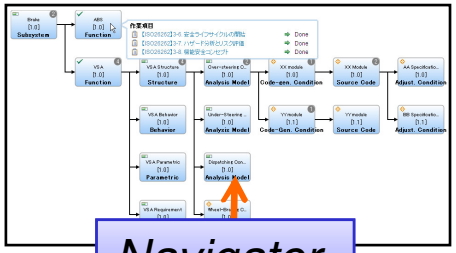
- Challenges for a federated integration architecture
  - Integrated views of data without data copying or export
  - Distributed configurations and baselines
  - Reporting and document publishing
  - Security
  - Performance
  - Each tool evolves with own time line



Agenda

- The Systems Engineering Challenge – Rational Perspective
- Engineering Lifecycle Management needs
- **Tool integration patterns for Engineering**
- Summary

Proposed Rational Engineering Lifecycle Management solution



RTC

RQM

Navigator

An “Engineering Integration Hub”

Rhapsody or RSA Design Management:
jazz-based storage and collaboration for models (new)

DOORS

Integrates with existing Rational data sources: e.g., DOORS, RTC, RQM, Rhapsody as well as 3rd party tools such as Mathworks Simulink

RDM

RPE

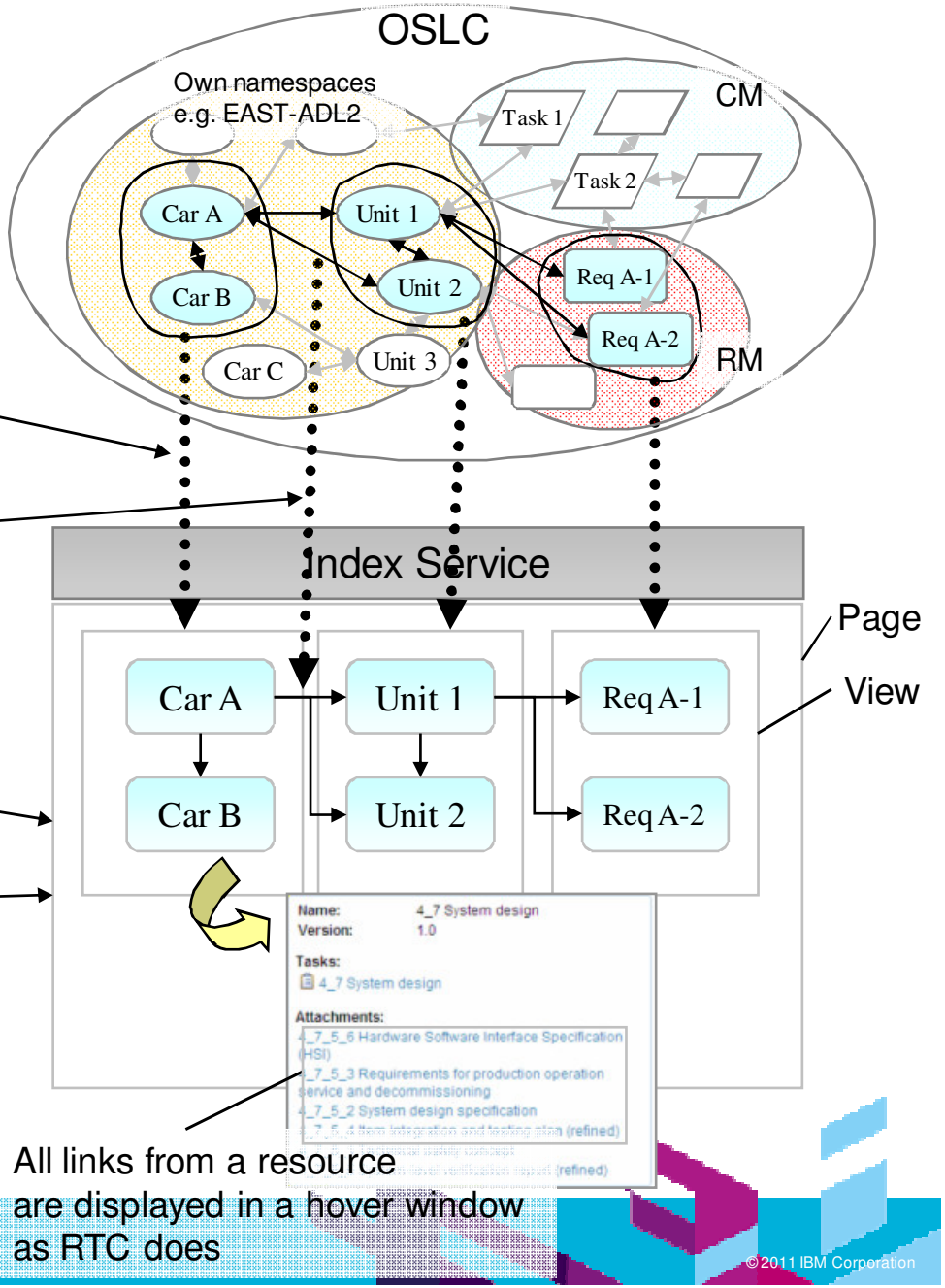
Insight

RPE and Insight integration for publishing and reporting

3rd party tools

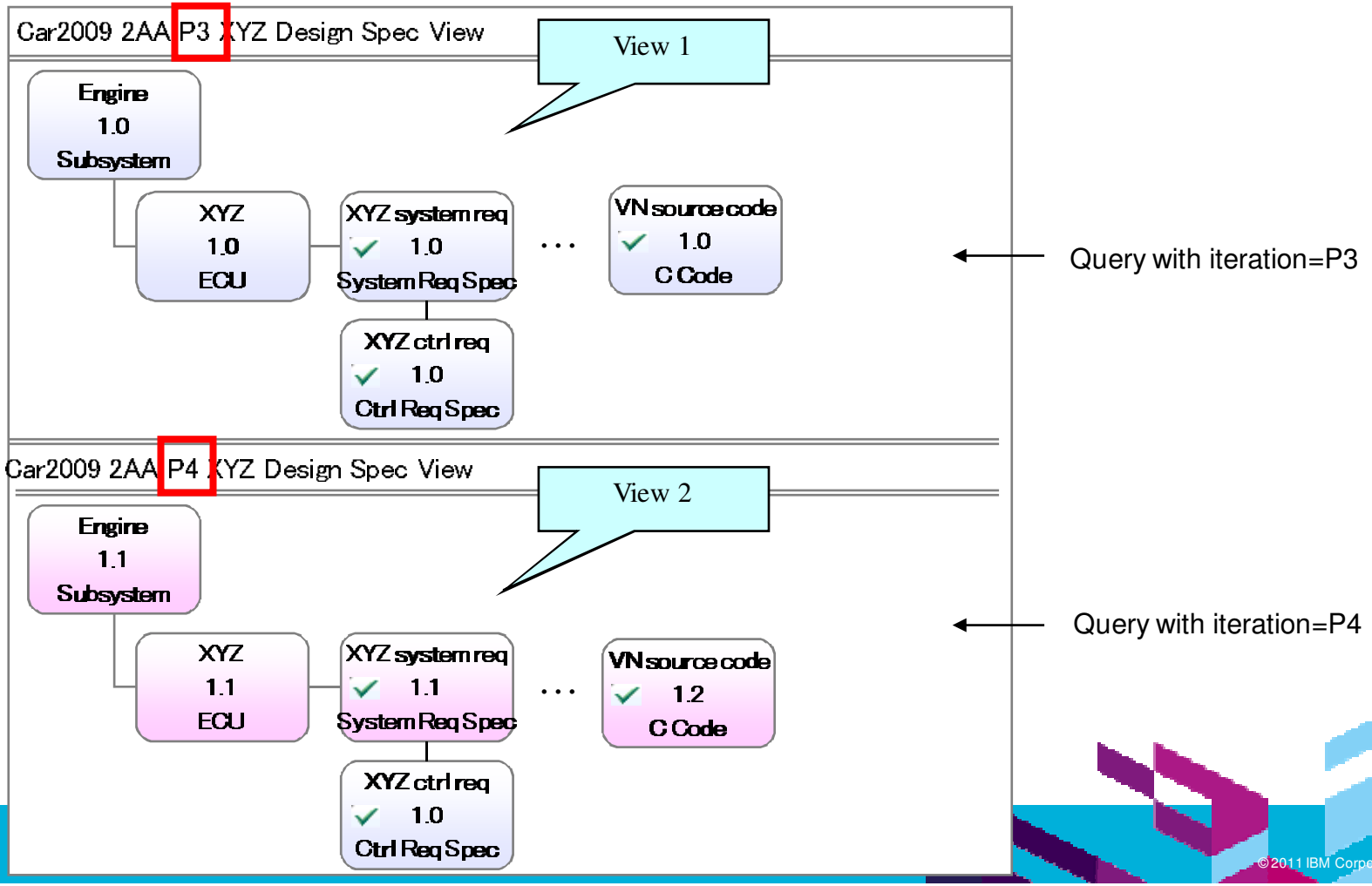
View Configuration (planned capability)

- Resource gathering
  - SPARQL
  - Search Keyword
  - Resource URL
- Link filtering
  - Show which link types in the view
    - e.g. show child link only
- Layout constraint
  - Table
  - Tree
  - Free form
- Behavior and Rendering customization
  - Menu and Actions
  - Background, Color, Font, etc.



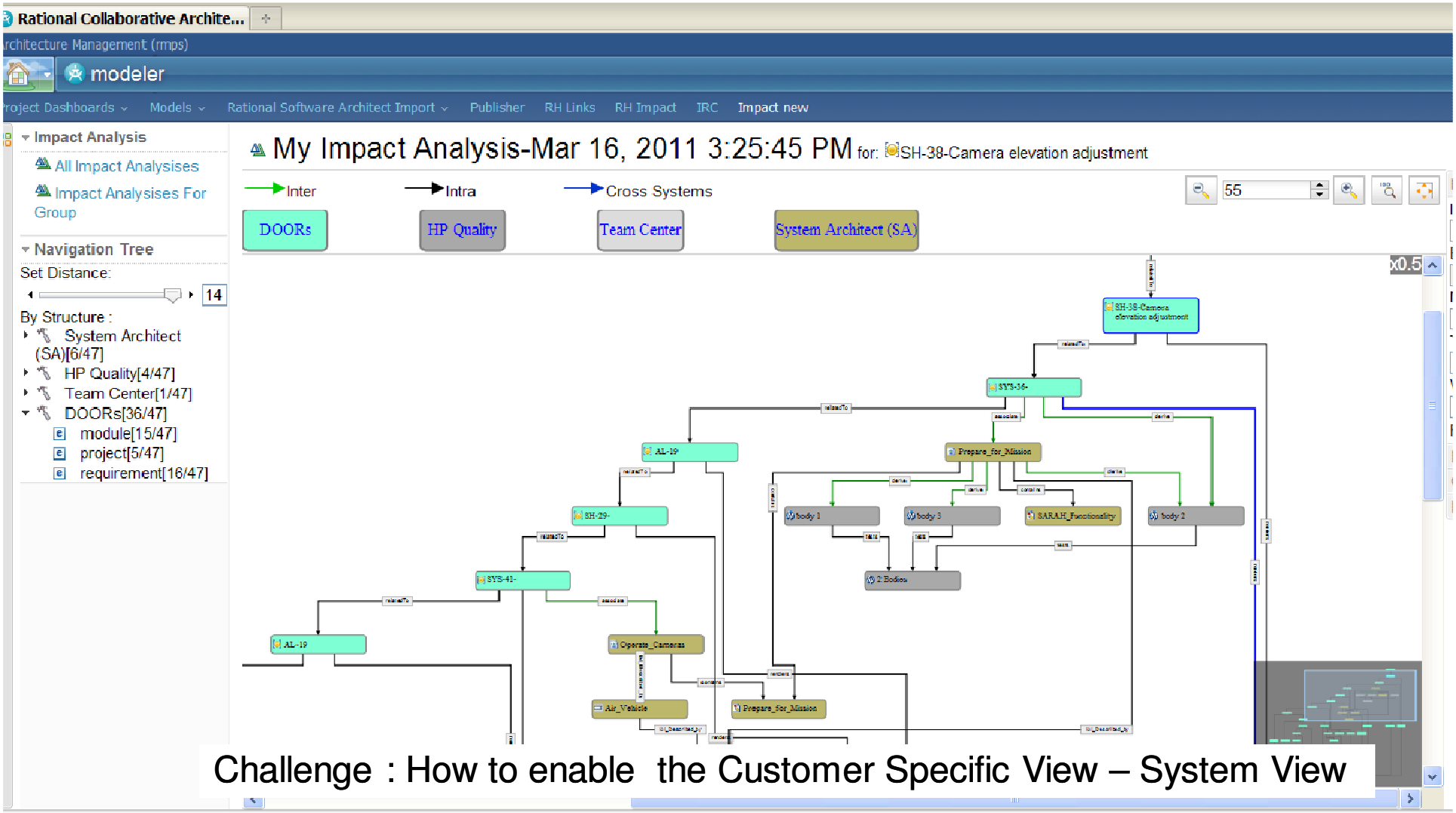
View Compare (example)

- Compare two views and show the differences (e.g. change color)
  1. Execute two SPARQL queries
     - (e.g. parameter *iteration = P3* for a query and *iteration = P4* for another)
  2. Compare the result sets



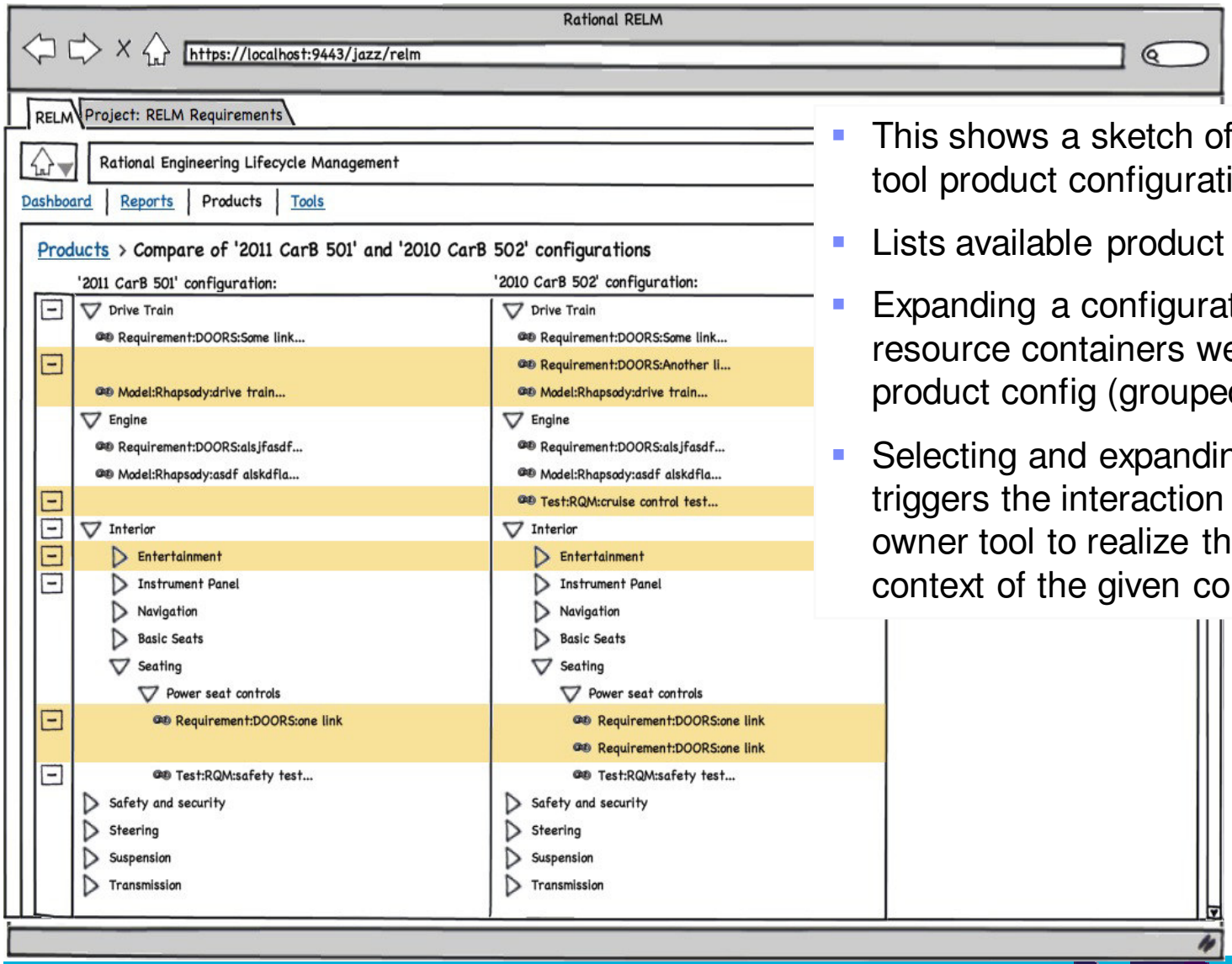
Impact Analysis UI example (from RelationshipHub project)

Structural view



Challenge : How to enable the Customer Specific View – System View

Product Configuration Example



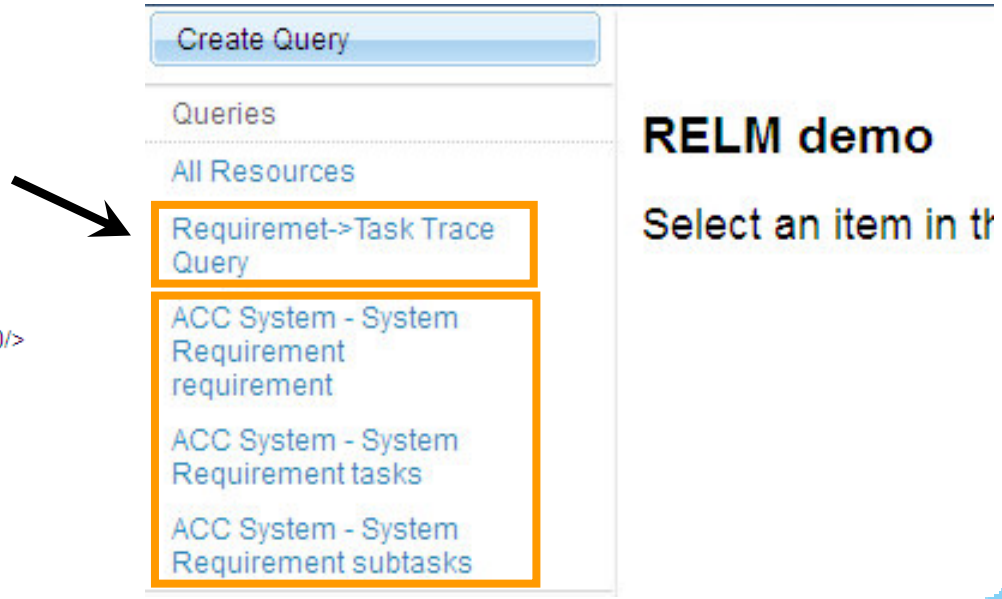
- This shows a sketch of a potential cross-tool product configuration UI
- Lists available product configurations
- Expanding a configuration shows the root resource containers we've identified in this product config (grouped by type)
- Selecting and expanding one of these triggers the interaction with the resource owner tool to realize the selection in the context of the given configuration

Query Page – Queries List

- Like Diagrams page, Queries page also has a navigation pane.
- Implementation: It retrieves queries from server by calling a REST service
  - https://localhost:9443/jazz/jfs/relmui/query/
- The server side implementation creates the list of queries by searching *.sparql files in a folder specified in relm.properties file
- In addition the list from *.sparql files, “All Resources” query is always added to the list
- Queries in configuration files which doesn’t have any parameter are automatically added to the list too

Example C:\relmdemo\sample.sparql

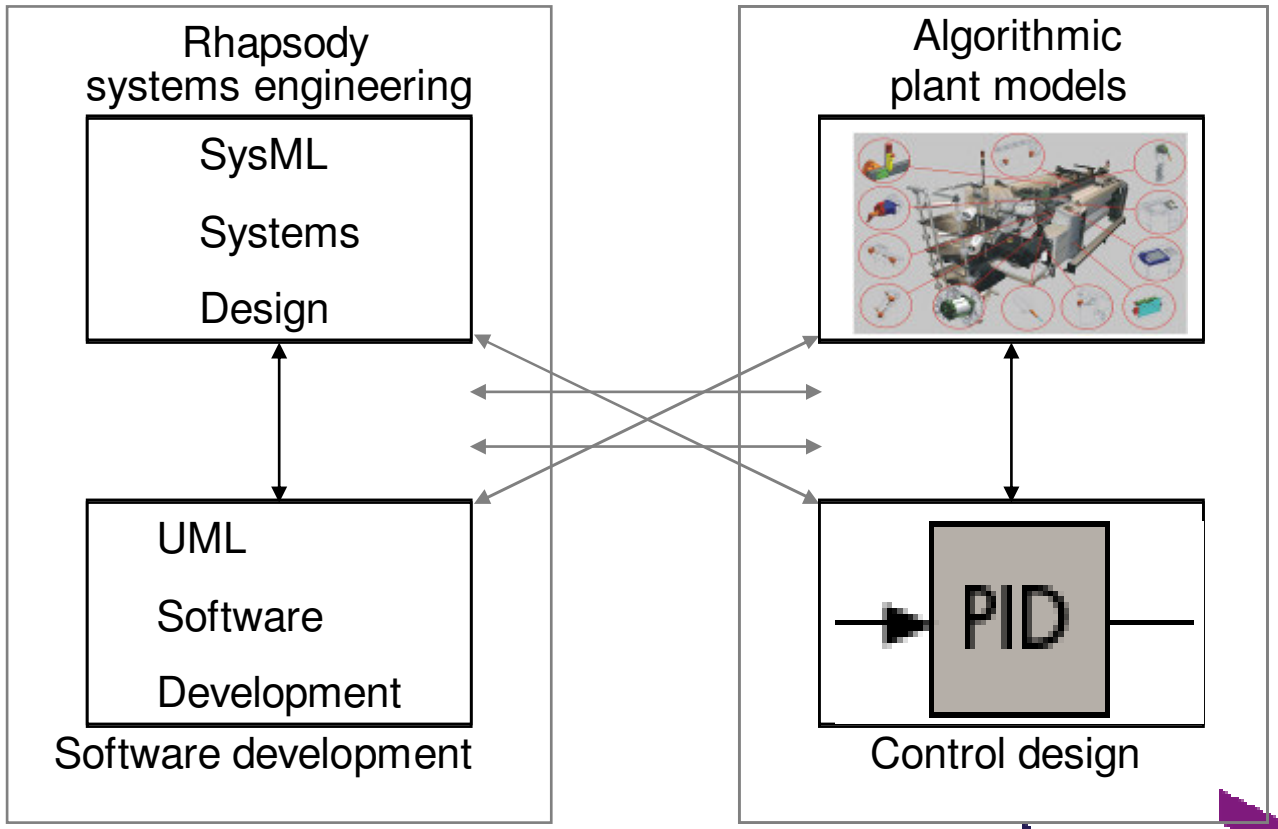
```
<queries>
        <query>
                <name>Requiremet-&gt;Task Trace Query</name>
                <sparql><![CDATA[

 PREFIX dc: <http://purl.org/dc/terms/>
 PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
 PREFIX oslc: <http://open-services.net/ns/core#>
 PREFIX calm: <http://jazz.net/xmlns/prod/jazz/calm/1.0/>
 PREFIX doors: <http://jazz.net/doors/xmlns/prod/jazz/doors/1.0/>
 PREFIX oslc_cm: <http://open-services.net/ns/cm#>
 SELECT ?requirement_title ?requirement_uri ?task ?status
 WHERE {
        ?requirement_uri
                dc:title ?requirement_title  ;
                dc:modified ?modified ;
                calm:implementedBy ?task .
               ?task oslc_cm:status ?status .
        }

        ]]></sparql>
        </query>
</queries>
```



Queries in configuration files

Rich Model Integration: Rhapsody and Simulink Planned capability

- Breakthrough combination of domain-specific modeling solutions
  - Provides an integrated multidomain solution for a multidomain problem using DSL



Additional Uses of Design Manager in RELM Context

- Other types of modeling tools could be integrated the same way:
  - -Electrical
  - -Electronic
  - -Other domain-specific/vertical
  - -Proprietary (even excel “models”)

- Design Manager will also be able to host new domain-specific models
  - Planned DSM toolkit
  - Useful for creating a domain-specific cross-tool information model to supplement the index and drive some of the RELM views/queries

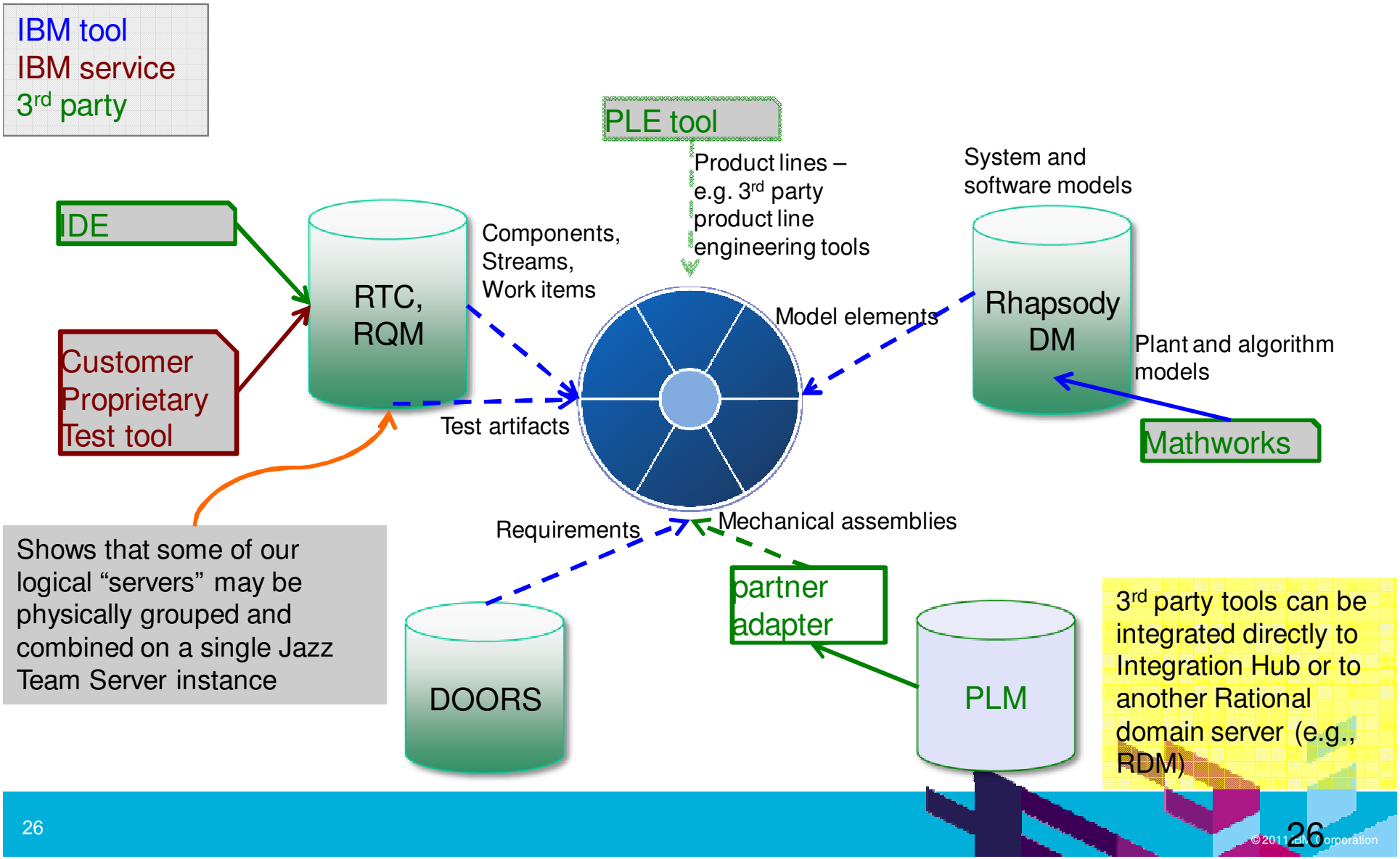
Engineering Process Integration

- Jazz cross-tool process enactment
  - Formulate an end-to-end engineering process across the lifecycle
- OSLC CM integration to index
  - Trace, query and visualize process information across multiple tools
  - Relate artifacts to process state
- RTC work item enhancements and process templates
  - Manage engineering plans and work activities using RTC
- 3rd party OSLC CM integrations
  - Integrate to process / change mgmt in other tools (e.g., PLM, MRO)

Planned Capabilities

Capability	Future Product
Support Systems Engineering Method	✔
Cross tool artifacts/models integrated view	✔
Cross tool models/artifacts bidirectional relations	✔*
Cross tool search and query	✔+
Impact analysis	✔
Cross tool reports and documents generation	✔
Cross tool baselining	✔
Cross tool product configuration	✔

Example engineering integration hub scenario



Engineering Lifecycle Management

Agenda

- The Systems Engineering Challenge – Rational Perspective
- Engineering Lifecycle Management needs
- Tool integration patterns for Engineering
- **Summary**

28

Rational solution for Collaborative Engineering Lifecycle Management

Capability to support effective teams – enhanced by central index and product "context" visibility

- Real time planning
- Lifecycle traceability
- In-Context collaboration
- Development intelligence
- Continuous process improvement

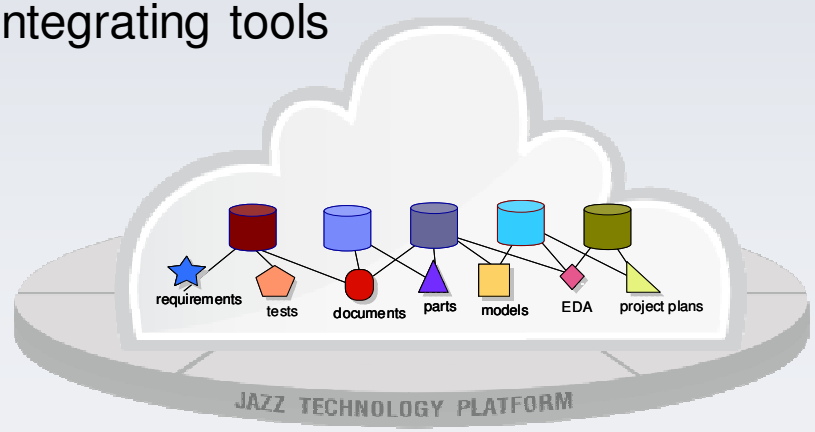
Connected data and tools

on a truly extensible, scalable and open platform

Key Directions

- Index of artifacts and relationships
- Resource navigation and query
- Cross-tool configuration management
- Cross-tool baselining
- Integrated multi-tool processes
- Open integration standards (OSLC) for integrating tools

**„open community.
open interfaces.
open possibilities.“**



30

www.ibm/software/rational

www.ibm/software/rational