



# OSLC PLM extensions proposal

For Oct 5<sup>th</sup> 2011  
V0.7



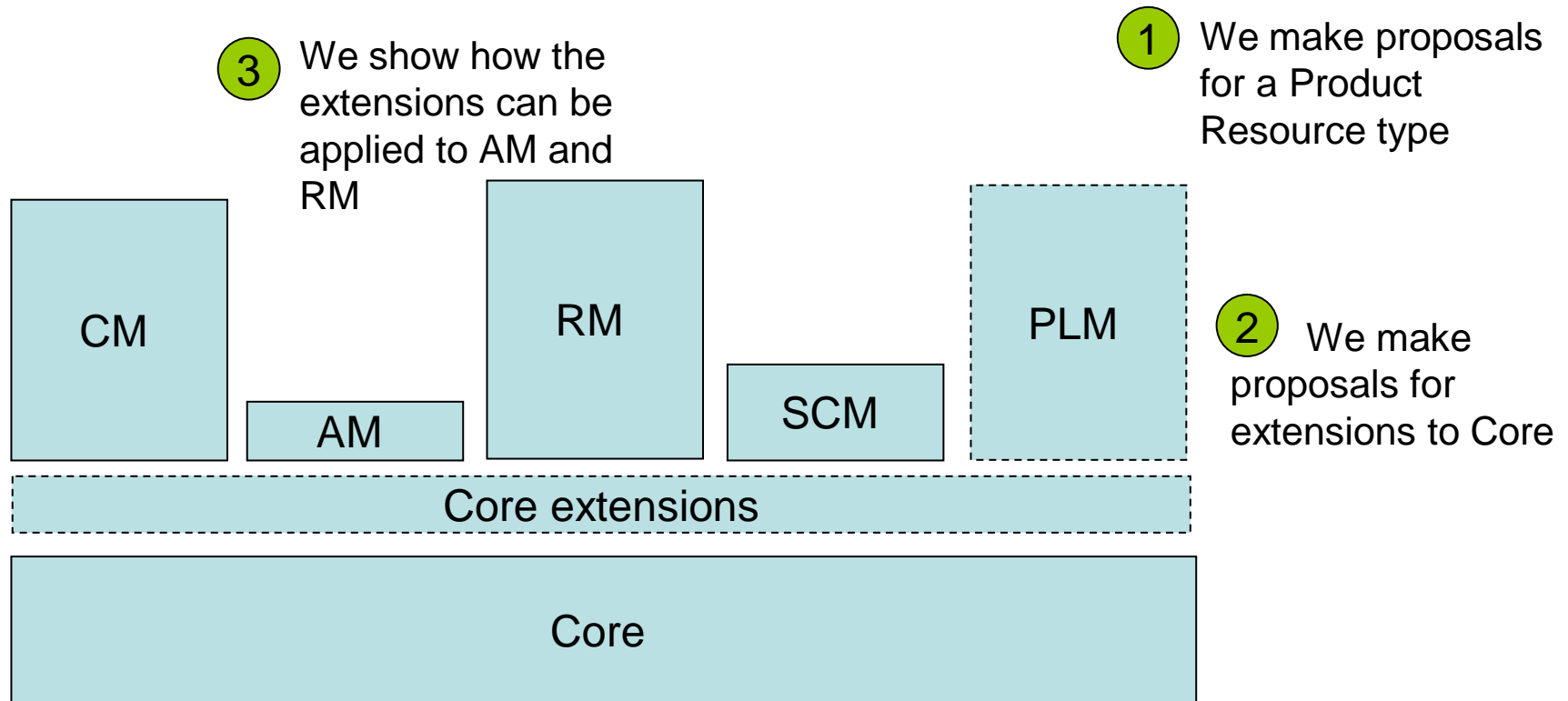
# Agenda

- Summary of the request to the Core WG
- Background to the proposal
- Proposal: Introduction to the required Product behaviour of an OSLC resource
- Next steps

# Summary of the request

- For the Core Workgroup to identify the sponsorship and make recommendations about the preferred approach to publish a workable draft set of PLM extensions by 2<sup>nd</sup> Dec 2011
  - What aspects should the Core WG handle ?
  - What aspects the other OSLC WGs handle ?
  - What should the PLM WG focus on ?

# Challenge for OSLC



NOTE: Any extensions to OSLC should allow an incremental adoption

# Background

- In today's OSLC specifications "product" is a mostly hidden concern
  - Many customer develop products
  - Many customers consume products in the context of SW development
  - Many customers develop applications that support products or their processes around products

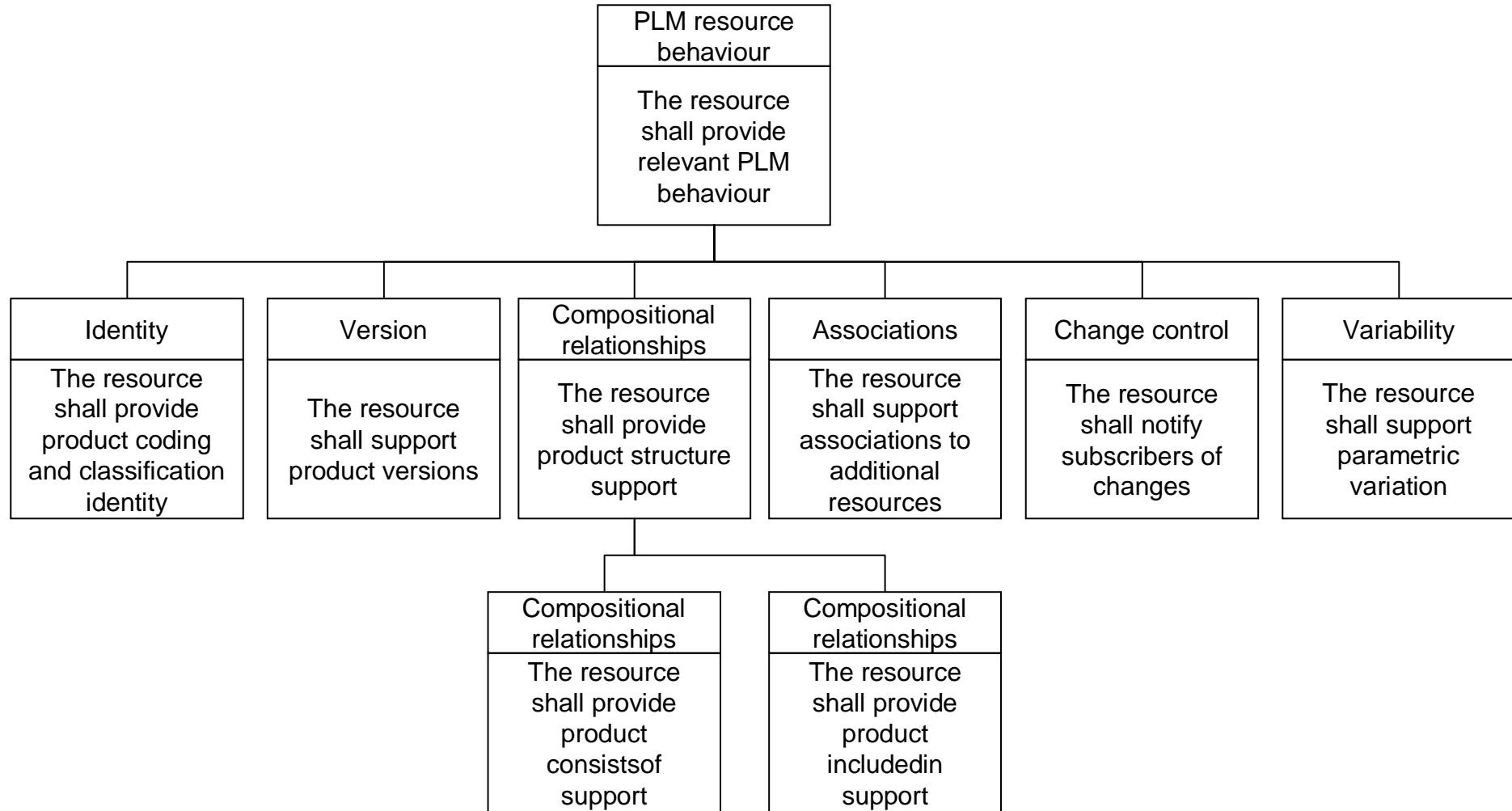
# Goal

- To extend the support for product lifecycle concerns with OSLC

# What does OSLC need ?

- A means of providing resources with product like behaviour
- A means of associating OSLC domain capability with product resources
- A means of transitioning resources from and to product resources

# PLM behaviour requirements of an OSLC resource





# Conclusions from Workgroup analysis

1. A version is an indicator of change
2. A view is the composition of a version
3. A release is a version that reaches some criteria

Therefore we assert that an item, item-version and view-definition construct is widely applicable

# Related OSLC concepts that could be applied in the absence of a Product Resource

OSLC Concept	Description	Source
Change Set	A set of changes described in terms of members (direct or indirect) (that are) added to, replaced in, or removed from some configurations	CM 2.0 Spec
Configuration	A collection,...set of resources, their states, and the links between them, is called a configuration	OSLC baseline proposal
Baseline	"immutable record of a configuration"	OSLC baseline proposal

PLM needs a focused concept that can also host  
product coding & classification,  
alternatives  
use of variant expressions to arrive at a versions (i.e. configurations)  
change management



# Introduction to the required Product behaviour of an OSLC resource

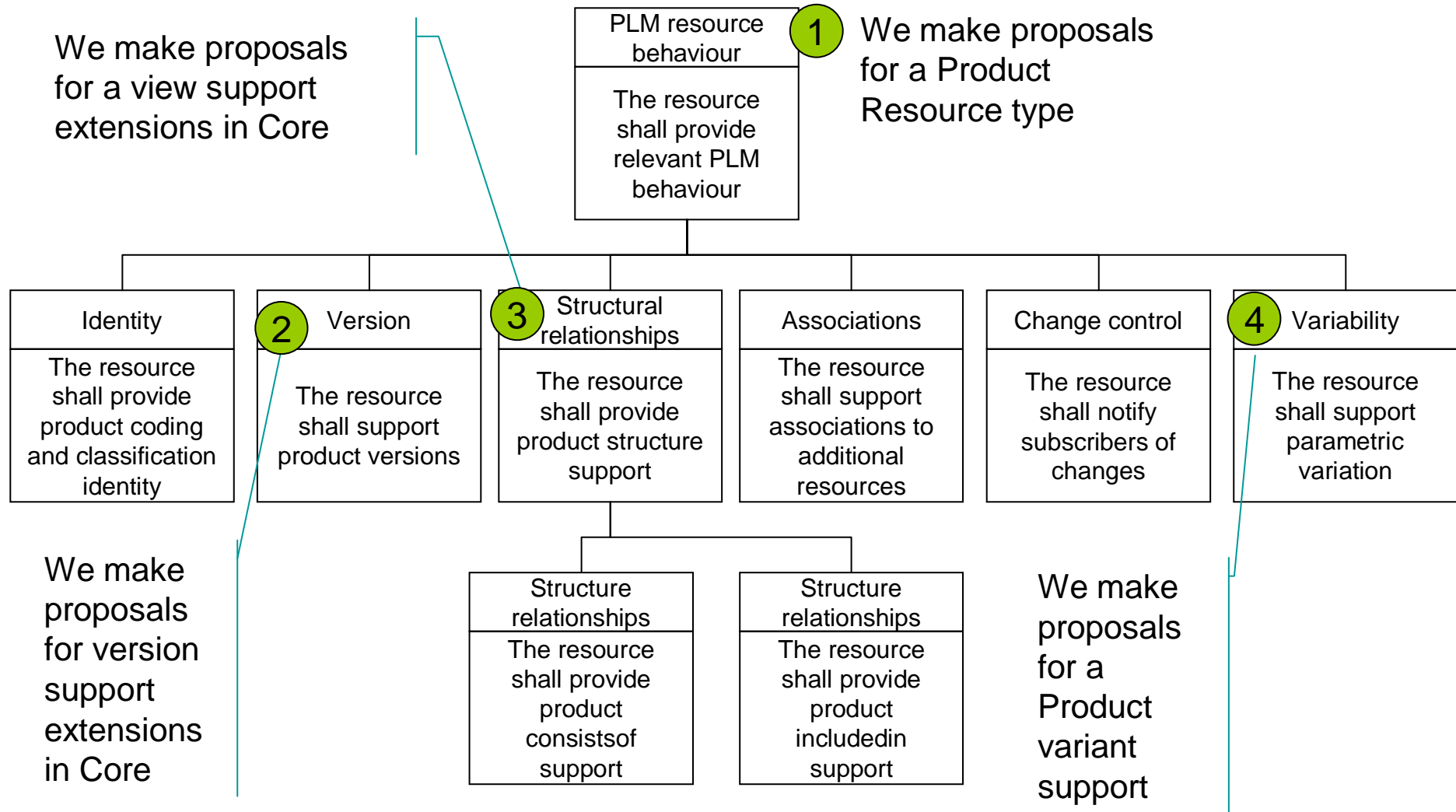
The full description is available here:

<http://open-services.net/bin/view/Main/Plm20SpecExtensions>

And examples are being built up here:

<http://open-services.net/bin/view/Main/PLMExtensionsOverview>

# PLM behaviour requirements of an OSLC resource



# Key concepts

- Product
- Product variation
  - Versions
  - Variants
- Product view

# The concept of a Product

- A Product is a very common and deceptively simple concept
  - E.g. A commercial unit of delivered function
- A Product provides a key unifying concept for business activity and enterprises
- Today it is a significant challenge to define systems products across the many concerns of contributors and consumers

# How should OSLC support the concept of product ?

- Through a specific Product resource type
  - > Current proposal
- Through extension of the resource definition for an existing OSLC resource, i.e. as a product “proxy”
  - > Current proposal

Show examples for AM and RM

# Product Resource

- Propose as a new resource type “Product”
- Extends OSLC Core resource definition
- Recommended use of selected OSLC resource definitions
- Optional application
  - Version handling
  - Domain views
  - Variant expressions to support effectivity and options
- Recommended approach to forming URIs from Application unique identities
- Recommended means to extend the terminology allowable
  - Additional namespaces
    - Industry standard e.g STEP
    - Proprietary e.g. PLMxml



# The concept of product variation

- The concept of what is a, or even *the*, product often varies greatly along the lifecycle
- In terms of it's generality or specificity
- In terms of it's precise composition
- In terms of representing or justifying it
- This applies to the variation of many lifecycle resources
  - product, requirement, implementation

# What are typical ways to handle product variation ?

Through a product “context” to support

- Indication of significant change
  - Business rules or ad-hoc criteria are used to signal significant changes of a product by way of a version or revision annotation
- Handling of alternative product capability and/or composition
  - Sets of determinants are used to resolve the product definition
- Handling of effectivity
  - When and where product variation become relevant

# How should OSLC support product variation ?

- Approach 1: Provide a generalised means to deal with multiple product resource, or it's proxy, contexts using managed sets of determinants
- Approach 2: Provide specific means for a product resource, or it's proxy, to handle
  - Versions
    - > Current proposal
  - Status
  - Variation by way of
    - Options
    - Effectivity
    - > Current proposal

# Summary of the version handling proposal

- Any OSLC resource can gain version support behaviour through extensions to Core
- Applicable to any OSLC resource, including the new proposed Product resource
- A base resource can have multiple version resources
  - hasVersion
- A version resource identifies its base resource
  - isVersionOf
- A version resource can indicate its maturity
  - replaces

<http://open-services.net/bin/view/Main/Plm20SpecExtensions>

# Summary of the variant expression handling proposal

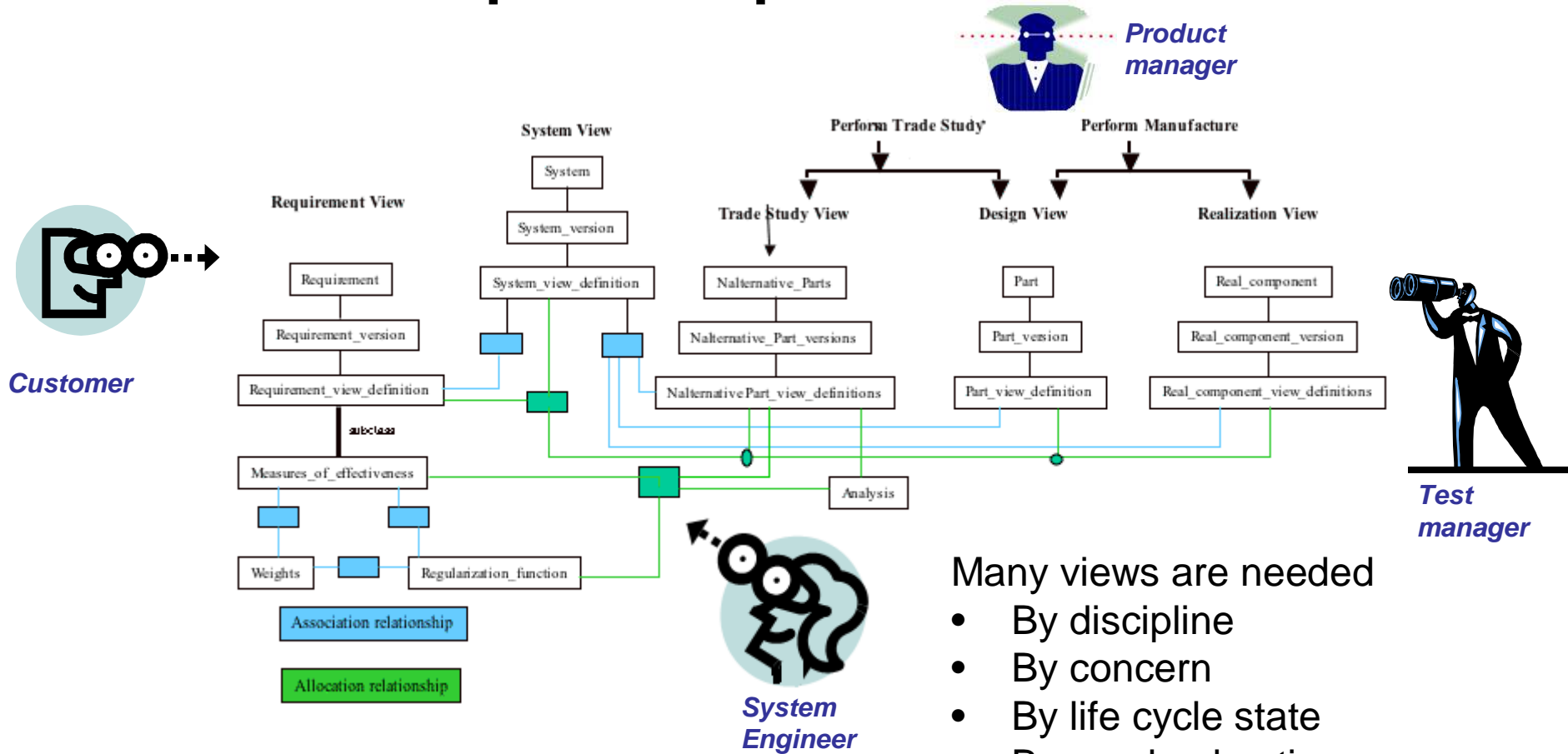
- Test of some criteria, if true then apply the criteria and yield 1 or n views
  - E.g. VariantExpression>MarketRegion=US
- Yields a Product view resource
  - i.e. a configuration



# The concept of a product view

- A view is a domain specific representation of a resource e.g. of a product or system, resolved by application of a context e.g. to a product or system

# Example of product views



- Many views are needed
- By discipline
  - By concern
  - By life cycle state
  - By resolved options
  - By effectivity

# How should OSLC provide support for product views ?

- Approach 1: Through a generalised means to apply context to a product resource, or it's proxy, to access or realise a view
- Approach 2: Through a means to locate a specific view of a product version resource
  - > Current proposal



# Summary of the view handling proposal

- Any OSLC resource can gain view support behaviour through extensions to Core
- Applicable to any OSLC resource
- A version can have multiple views
  - View types can be handled via subject
- A view resource has members
  - hasPart
- Propose that part of is achieved through inference

<http://open-services.net/bin/view/Main/Plm20SpecExtensions>

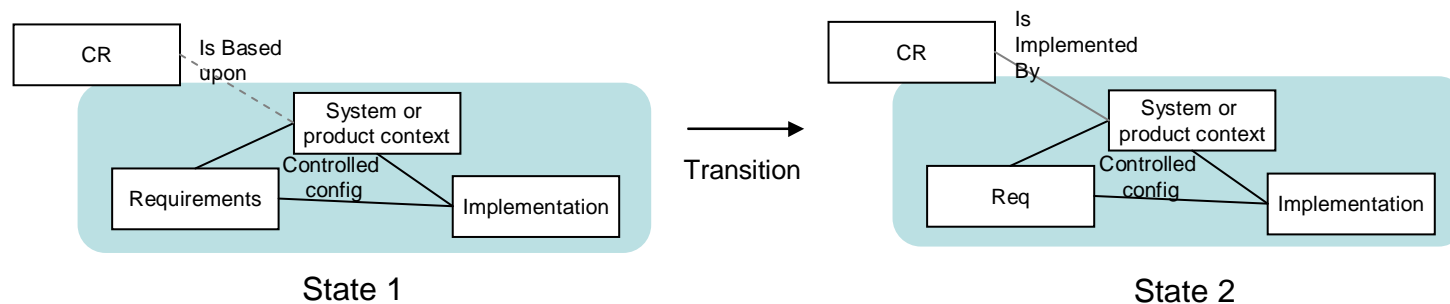
# Pros and cons of the PLM extension proposals

- Pro
  - We can get started
  - Relatively simple
  - Consistent with Existing Product Data Standards i.e. ISO-10303 - STEP, OMG SysML
- Con
  - Limits the current applicability to specific contexts/views
  - Lacks a generalised approach
  - Will need to build out with approaches for effectivity and effectivity configuration management

# Putting the proposals into practice

# The main PLM scenario is a typical industry case

- Dear Systems Engineer please “implement a change to a system product” that is at some defined state, and make it available at a new state



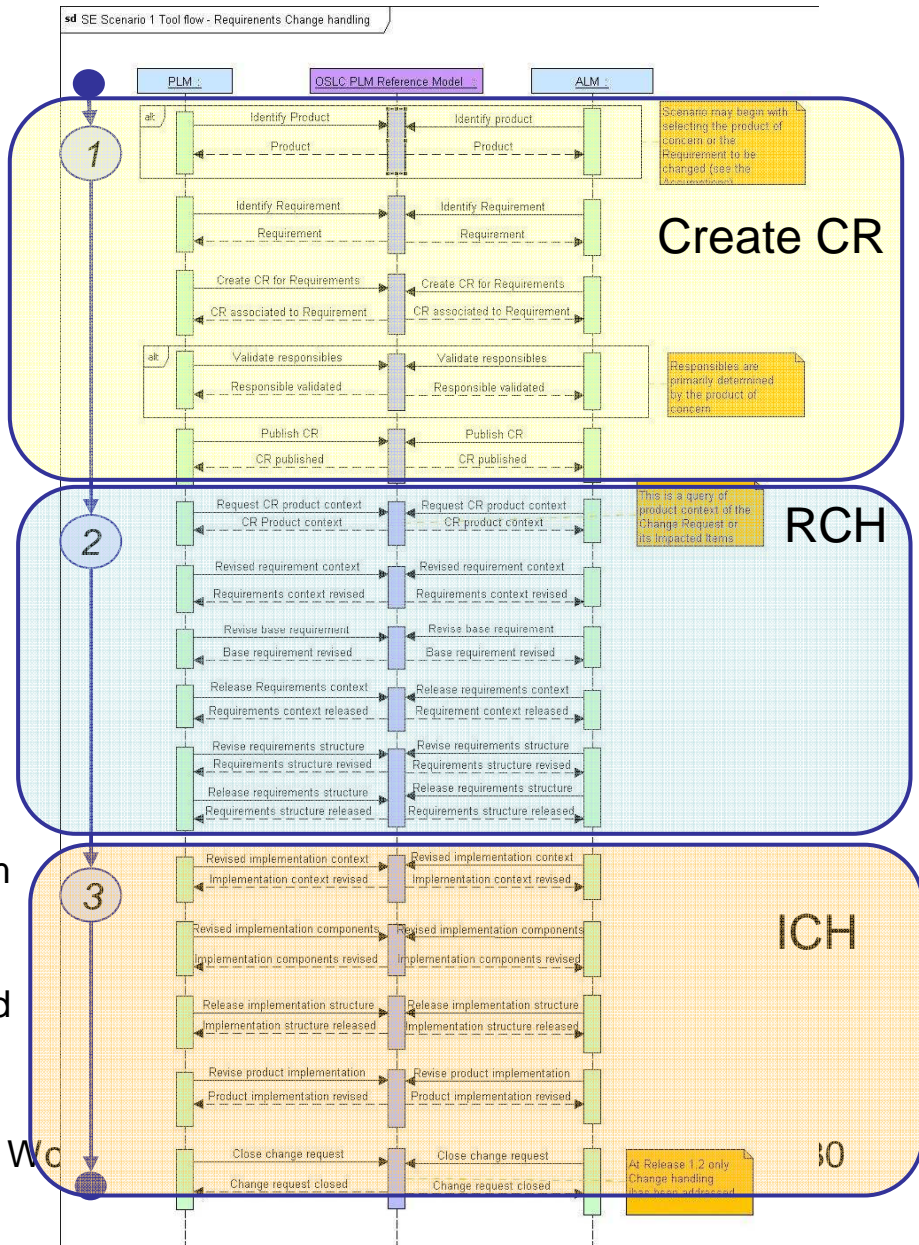
# The OSLC PLM reference model shows the scenario needs:

- Representation of a product resource
- Association of OSLC resources with products
  - CM Change Request
  - RM Requirements
  - AM Resources
- Support for variation
  - Versions, variants
- Support for views

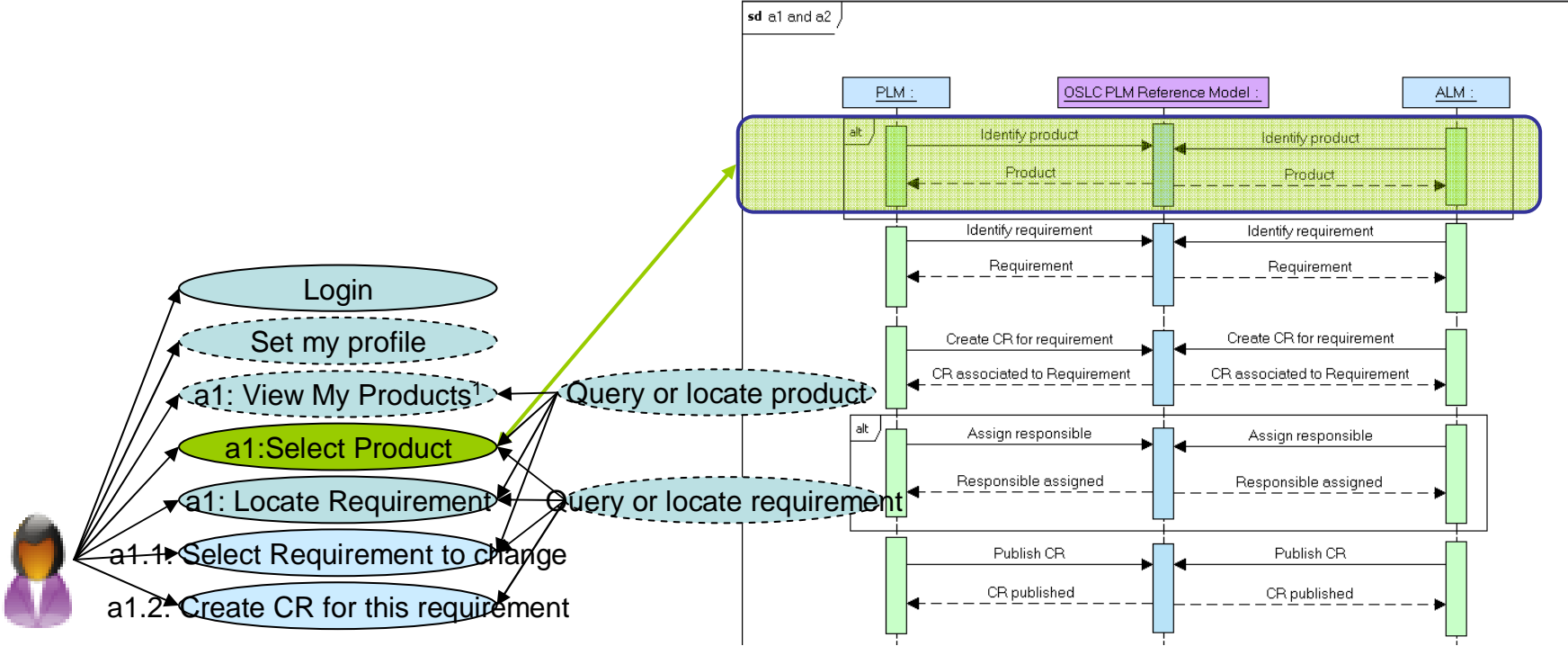
# Scenario sequence

A system responsible, i.e. a Systems Engineer, in a Business unit, needs to respond quickly and accurately to requests for product and system changes to meet responsiveness and cost objectives

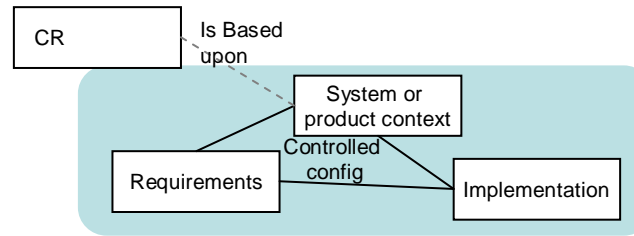
- 1 create the change request for product update and make available to the responsables
- 2 prepare an update to the system requirements to fully specify the change request, working on the appropriate areas, re-using relevant requirements and calling upon other contributors, as needed, to meet the business objectives
- 3 prepare and oversee an update of the full system definition as an implementation of a solution to the change request, working on the appropriate areas, re-using or designing relevant content and calling upon other contributors, as needed, to meet the system objectives



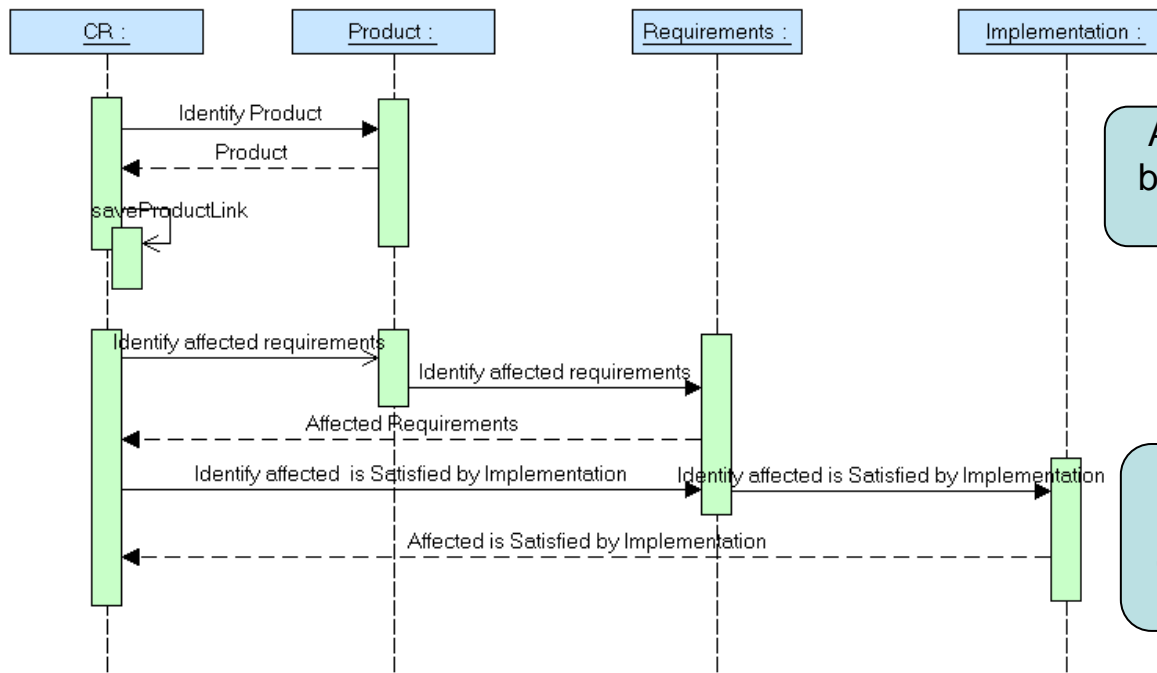
# Example High Level use case



# Summary of basic support



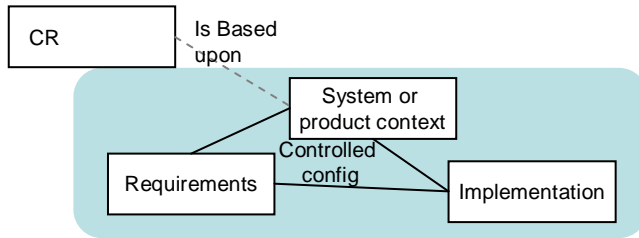
sd Basic resource behaviour



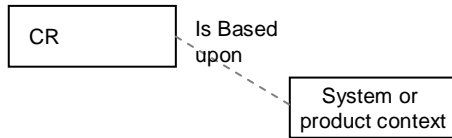
An OSLC CM resource can be associated with an OSLC Product Resource

An OSLC Resource can be used to locate an OSLC RM or AM resource, typically via it's versions and views

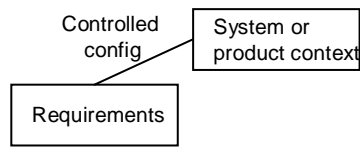




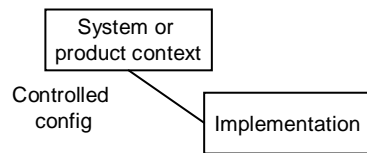
# Overview of the application of OSLC Specs overview to the scenario



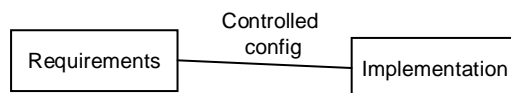
1. Product resource supports resolution of variants including revisions
2. Product version and view resources support variations
3. CR resource should support tracksChangeSet for a Product version or view
4. CR resource could use subject term as a Product resource identifier



1. Requirements resource could support isVersionOf a Product
2. Requirements view resource could support isVersionOf a Product version
3. Requirements view resource could support isVersionOf a Product view
4. ISSUE: Need to find requirement isRequirementFor by inference
5. Product resource could have an isElaboratedBy Requirement relationship



1. AM Resource could support isVersionOf a Product
2. AM Resource view resource could support isVersionOf a Product version
3. AM Resource view resource could support isVersionOf a Product view
4. AM Resource can support a Linktype, say Implements, a Product resource



1. Requirements resource can support isSatisfiedby to an AM implementation resource
2. AM Resource can support a Linktype, say Satisfies, an Requirements resource



# Discussion around the request

- For the Core Workgroup to identify the sponsorship and make recommendations about the preferred approach to publish a workable draft set of PLM extensions by 2<sup>nd</sup> Dec 2011
- The Core WG to handle...
  - Promote for feedback and contribution
  - Community newsletter
  - Sponsor closure by Dec 2nd
- Other OSLC WGs handle
  - Align for these concerns : .....TBD
- The PLM WG to focus on
  - Handle Q&A
  - Build up evidence, examples
  - Build up Wiki support
  - Input to Community newsletter
- Next report back in 2weeks



- Thank you to all the PLM Workgroup members who have contributed
- This summary was assembled by Gray Bachelor
- Particular thanks to
  - Mike Loeffler, GM
  - Hiroaki Nakamura, IBM Research
  - Hisashi Miyashita, IBM Research
  - Andreas Tsiotsias, IBM