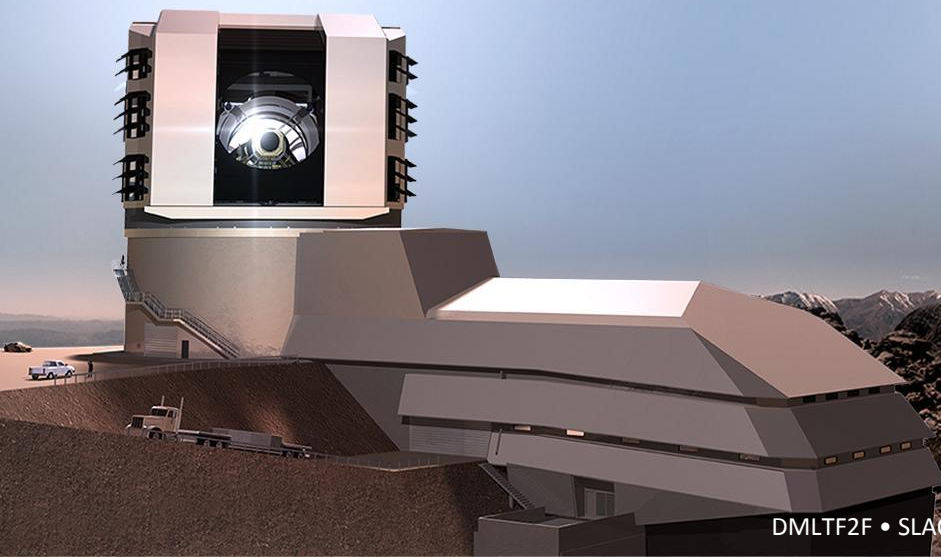




# Detailed Product Tree Gabriele Comoretto Configuration and Release Manager

DMLT Face-to-Face  
October 28-30 2019



- Product Tree Scope
- Current Status
- New document
- Document structure
- Consistency and Completeness
- Conclusions
- Improvements

The Product Tree provides a schematic view of the products we are working on.

The products are the project's components that fulfil the requirements:

- These components are not defined by the Product Tree
- The product tree is just a schematic way to view them

The scope of the Product Tree is:

- Give visibility to all products
- Make everybody aware of them
- Agree on the naming.

Why is the Product Tree important?

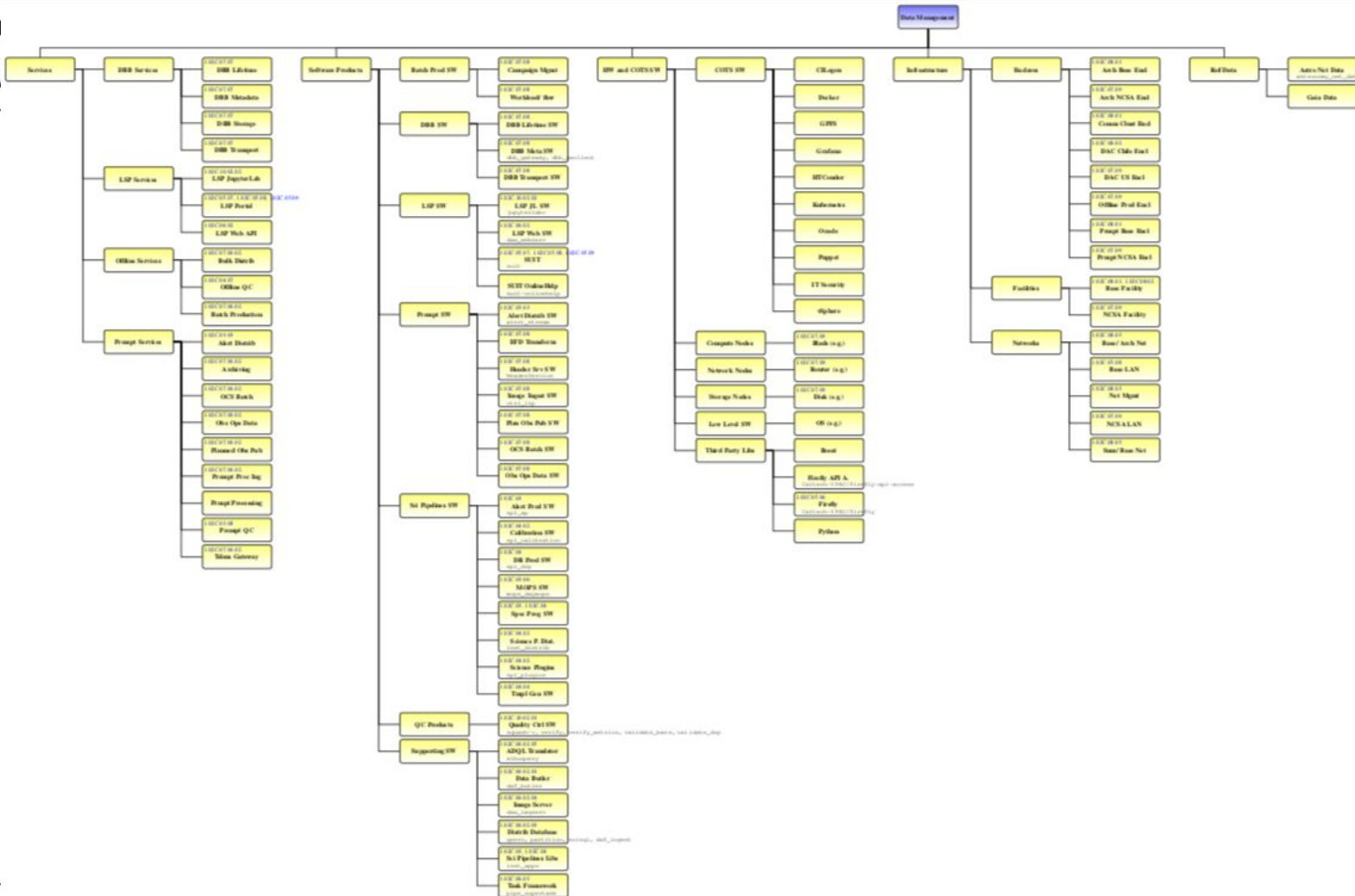
- Internally: Help people inside the project
  - Gives a common set of definitions
  - Global understanding
  - Help newcomers
- Externally: gives a visible idea of the (huge) work done

The Product Tree is not showing anything new:

- It just provides a schematic view of the products that already exists in the project.

- The DM Product Tree
  - Is a graph attached to LDM-294:
    - <https://ldm-294.lsst.io/ProductTreeLand.pdf>
  - Only operational products are included
    - Those products that are used to produce LSST data
  - It groups the products per category
    - No functional, build or runtime dependencies are visible in it
  - Each product is NOT described in detail.

- The Product Tree is
  - Generated from the MagicDraw DM logical model.
  - Those components are defined a priori, and serve other purposes
    - System engineering modelling
    - Requirement traceability
    - Integration with other parts of the project (non DM)
- LDM-148 provides descriptions on relevant DM areas, services or groups of products.



- Currently, when a new version of the Product Tree is created, it is not easy to answer the following questions:
  - What are the changes?
  - How we review those changes?
- These questions has been asked this summer when preparing the new LDM-294 for the JSR:
  - <https://jira.lsstcorp.org/browse/DM-20838>
  - See also DM-17439 (the topic is older, summer 2018?)
- Also: the information provided in the product tree, is very minimal
  - Some product names or abbreviations may be misleading without a bit of explanation



Having a new document can address all points from previous slide.

This imply the following benefits:

- More clarity
- More information provided in a single place
- Keep everybody aligned
- Less misunderstanding
- Fewer problems
- Quicker development
- Fewer maintenance issues
- Better organization (more efficient)

- DMTN-104 provides for each product in the Product Tree:
  - More detailed information
  - Easy to access information (only few people can access MagicDraw)
  - Easy way to review changes (documentation workflow using GitHub PRs)
- MagicDraw is the central repository of information, maintained by DM Arch.
- Moreover:
  - We can document additional products (non operational, or LSST generic)
  - Integrate the information from GitHub
  - Document other dependencies
- Draft at <https://dmtn-104.lsst.io/>

- The document can be fully generated using MagicDraw REST API
- No effort is required from the DM team to write the document
  - However information in it needs to be reviewed by the product managers and owners

- Introduction
  - Includes scope, definitions, and approach
- Top Level Product Tree
  - DM products required to fulfill the project requirements  
(Operational Products)
- Supporting Products
  - DM products required to build, work with, and maintain the Operational Products
- Low Level Product Tree
  - These are the GitHub repositories that implement the products (if relevant)

- The Products in MagicDraw are grouped in **folders**, using a functional characterization
  - Services
  - Software
  - Hardware and COTS SW
  - Infrastructure
  - Reference Data
- These **folders** may be considered also products, but in principle only the components (leaves in the tree) are real “Products”
- MagicDraw folders are rendered in subsections/subsubsections/paragraphs etc
- The Products:
  - Are rendered in tables
  - Do have dependencies (used in, uses)

## 2.2.3 LSP Services

Manager	Owner	WBS	Team
	Gregory Felsmann	Dubois-	
( Short Name: LSP Services - Acronym: LSPSRV )			

This group of services provide an exploratory analysis environment for the end user. These services are usually referred as "Aspects", identified as follows:

- LSP Portal Aspect
- LSP JupiterLab Aspect
- LSP WebAPIs Aspect

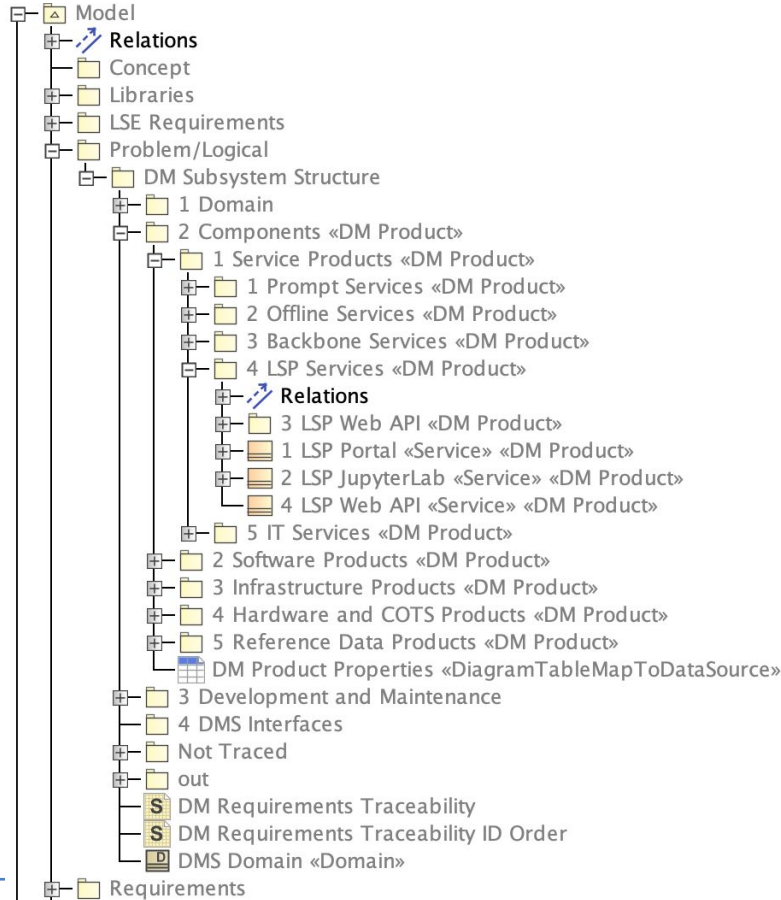
These services permit external users access to project resources, such as SW.

More details available in [DM-542](#) and [LSE-319](#)

<b>LSP Portal</b>				(product in: LSP Portal )
Manager	Owner	WBS	Team	
Xiuqin Wu	Gregory Felsmann	Dubois-	1.02C.05.09	
			1.02C.05.08	SUIT
			1.02C.05.07	
( LSP Portal - LSPPRTL_SRV )				

This service provides Web-based query and visualization tools for all the LSST data products.


Uses:	Used in:
<a href="#">SUIT</a>	<a href="#">DAC US Enclave</a>
<a href="#">SUIT Online Help</a>	<a href="#">DAC Chile Enclave</a>
	<a href="#">Commissioning Cluster Enclave</a>



Specification of Service 1 LSP Portal [Read-Only]

### Specification of documentation

Write documentation for the selected Service.



---

#### Documentation

- 1 LSP Portal
  - Documentation
  - Navigation/Hyperlinks
  - Usage in Diagrams
  - Usage In
  - Attributes
  - Ports
  - Operations
  - Signal Receptions
  - Behaviors
  - Template Parameters
  - Inner Elements
  - Relations
  - Tags
  - Constraints
  - Ports/Interfaces
  - Properties
  - Instances
  - Traceability
  - Allocations
  - Language Properties

HTML

This service provides Web-based query and visualization tools for all the LSST data products.

Delete

Close
Back
Forward
Help

- All products shall have an **Owner**
- All MagicDraw folders shall have a **Manager**
- The managers needs to ensure that:
  - She/he is the right person to take care of that folder
  - The description of the of the folder is correct
- The owner of each product needs to ensure that:
  - She/he is the right person to take care of that product
  - The description and metadata of the product are correct
- In case any correction is needed, open a Jira Issue to DM Architecture.



- Facilities: are composed of Enclaves
- Enclaves: are composed of Services
- Services: are implemented by DM Software Products or COTS
- DM Software Products: shall have only one GitHub repository (top level)
  - See software product definition in DMTN-106
  - Dependencies to 3rd party libraries should be derived automatically from GitHub (not yet, but will be)
- COTS and 3rd party libraries: shall have a link to a documentation page
- DM Architecture shall be in charge to ensure completeness and consistency
  - Owners and managers involvement shall be minimal

So far DMTN-104 document will:

- Provide detailed information for each product (and groups of products)
- Facilitate the access to the products information
- Facilitate the review (and correction) of the information

Bottom line:

- MagicDraw will be the repository of the product's information
- Other documents (like LDM-148 or LDM-294) will get the same information

DMTN-104 document could be completed in 6 month time.

Question:

- Should this be an LDM document?

- Some look and feel aspect may still need to be improved
- Add Documents (where available): for example specifications
- Add Requirements related to the components in the model
- The majority of the products have insufficient information in the corresponding MagicDraw components
  - First DM Architecture to complete a draft
  - Then all Owners and Managers should assess and provide feedback
- The low level product tree section is still empty
  - Implementing this section may generate some requests for improvement to the GitHub repository owners
- Integrate NCSA model (?)

- The provided list of components seems to be a good candidate to maintain the Jira DM Component list (DM-17440)
- This may require some work:
  - Additional information to be added in MagicDraw
  - Integrate DMTN-104 or MagicDraw with Jira(scripting)
- Generate LDM-148 using the same Rest API
  - Other documents (or parts of) can be generated also from MagicDraw
- The products are evolving continuously... in the long term, it is important to be able to keep this information up-to-date with the minimum effort.
- Generate the requirements specification using the same approach (pandoc)
- Export the approach to other project subsystem

Questions?