Requirements Engineering @ ASML

Experiences & Challenges

Dana Christensen
PL/TL/CL Requirements Engineering Competence

January 27th, 2022
Online

Contents

Questions to Consider

In your company:

How do you manage changes to requirements (and designs)…

…during development?

…later in the lifecycle of an existing design or product?

…when the change is in a design that is used in multiple products?

Imagine: you are deep into development (*e.g.*, detailed design) and a requirement changes…

- How do you assess the impact?

Imagine: you are deep into development of a part and a controlling requirement changes at system or function level…

- How do you know that a change has occurred?
- How does the system engineer or function architect know that you are already implementing the *previous version* of the requirement?

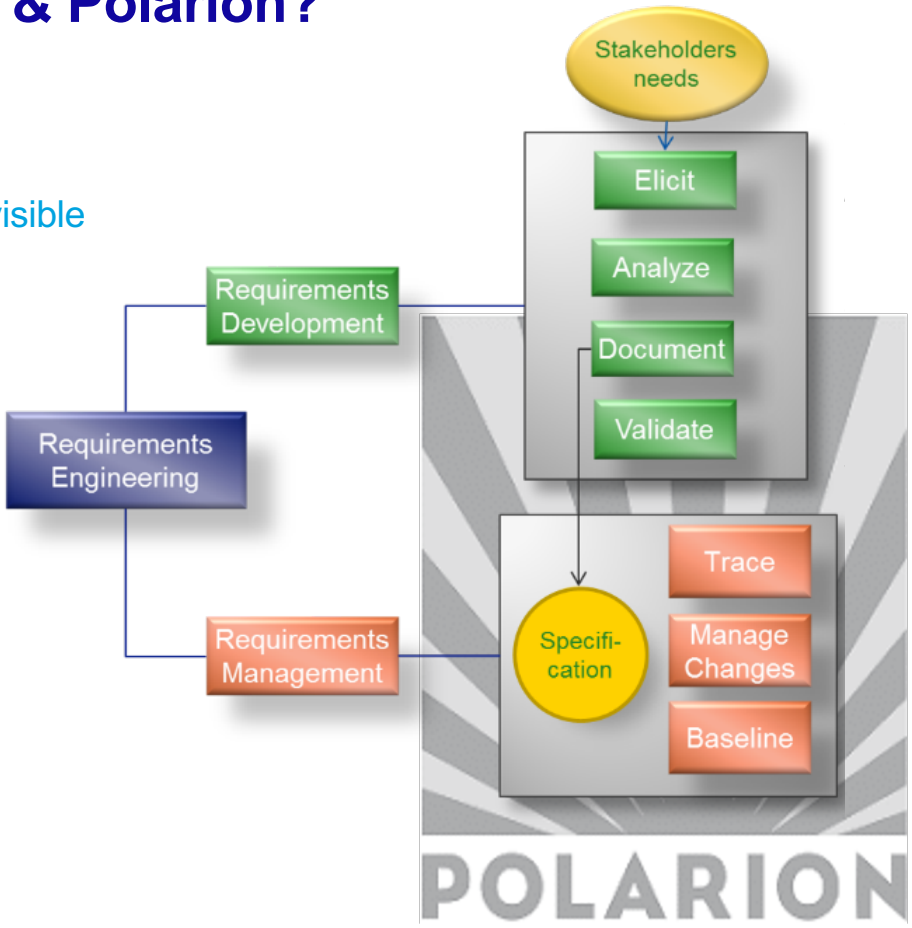
How do you ensure that your future architects and engineers understand where the requirements came from and the reasoning behind the design choices made today?

NOTE: ASML chose for Polarion as req mgmt tool, but every company should do their own tool selection based on their own needs and req eng process. A tool must support your process, it should not define the process. Note: as you gain experience, the tool may inform your process (like a learning loop), particularly in how you *implement* the process.

Summary: Why Requirements Engineering & Polarion?

Traceability is Key

- **Transparency on quality & completeness**
  - Missing, undocumented or orphan REQs & DDs become visible
  - Ensures clear ownership
- **Impact analysis** easy & fast in Polarion
  - For proposed changes
  - To prevent issues from unmanaged changes
- **One requirements database** in Polarion
  - One central, searchable point for information, up-to-date
  - Improved traceability of REQs & DDs
  - Managing changes easier & transparent; complete version & change history
  - Improved communication of changes in REQs & DDs
- **Improved initial quality**
  Polarion automates error-prone tasks (*e.g.*, unique IDs)



Polarion helps us to better manage complexity and facilitates concurrent engineering

Benefits are clear: easy sell, right?

No. (At least not in ASML)

Brief timeline:

2011 – Start of project; pressure to start immediately with tool selection

2013 – Tool selection abandoned; project restarted with focus on RE process

2016 – Basic RE process agreed in principle with a multidiscipline group of SE and D&E architects

2017 – Tool selection re-started; piloting of process (using Word) to get feedback and make improvements

2018 – Polarion chosen; installed and configured; launched in October

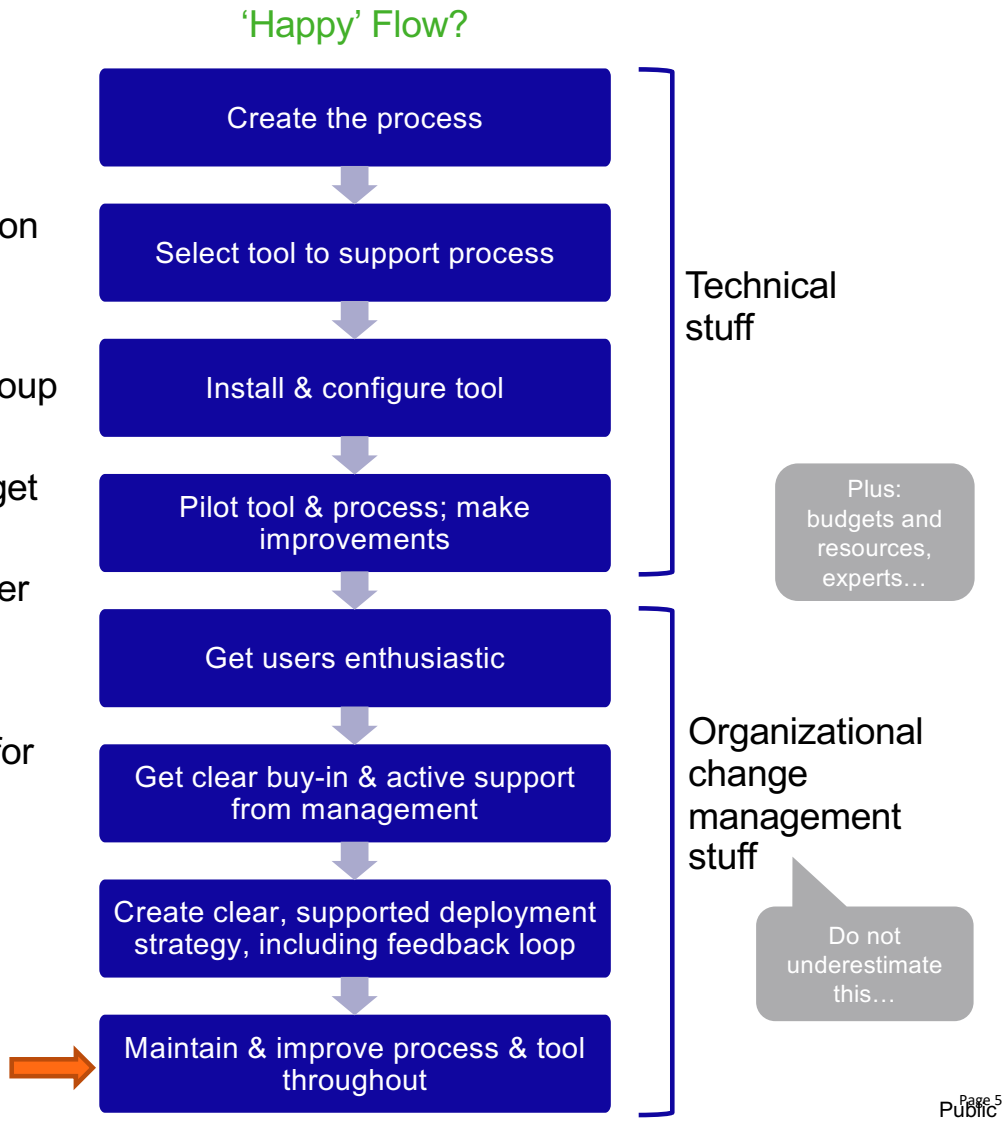
2019 – Start of extensive piloting of Polarion and process together; completed in December

2020 – (Finally) received GO and support from (D&E) management for formal deployment

2021 – (Finally) convinced management to take active role in deployment to make faster progress

2022 – Goal is to complete deployment in EUV

2023 – Start deployment in other business lines (DUV & APPS)



Deployment requires a lot of support for the architects & engineers

Requirements Engineering Process

Training

CBTs

Live workshops & trainings

Process Key-Users in the FCs (& some depts)

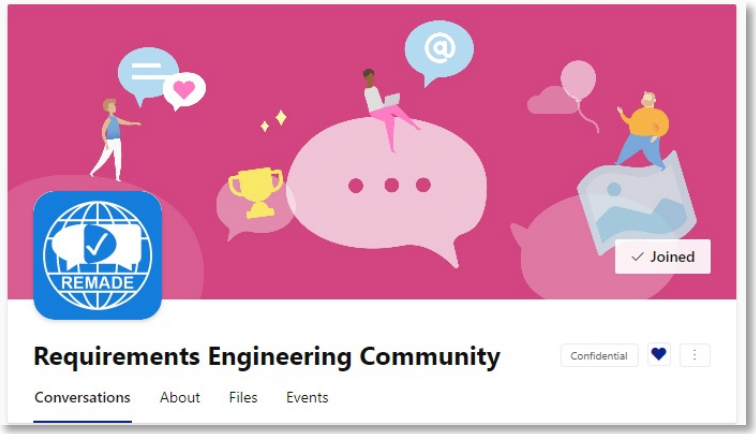
Active role in implementation and support of the RE process in the FCs

POLARION

Polarion Training (CBTs)

Polarion Key-Users

Day-to-day support for use of the tool



Yammer Community of Practice

Internal social network

Ongoing challenges go beyond just the RE process & tool

- ASML is large; D&E is ~15000 people, of which ~7500 write requirements or design specifications
- Even with a scope currently limited to EUV, my team of <10 FTE is supporting ~3000 people
  - Key-user networks are critical; clear trainings and support materials are essential
- Time-to-market is king; getting people to dedicate time to training & using a new tool can be difficult
- Transition from Word documents to sets of requirements in a database is not easy for many people (despite the document-like user interface)
  - Working in a database is not the same as working in a word processor – it is far more restrictive (which is necessary to achieve the main benefits)
  - Many still think of documents as program deliverables → a document is just a type of container for a consistent set of requirements
- 'Early adopters' are pushing for new & improved functionality, while the 'late majority' and 'laggards' are pushing to make Polarion behave more like Word
- The way ASML concurrently develops products that have common parts makes configuration management difficult, which also affects how the RE process and tool can be set up most efficiently for users (*e.g.*, variant management, re-use, traceability across products, ...)
- How to best involve co-developers in this transition?

Concluding remarks from the ASML perspective

- Focus of this discussion was on requirements *management* and deployment of RE process and tool; still so many things not mentioned here, *e.g.*, requirements *development* (& requirements quality), the link to verification & validation, etc.
- Requirements engineering touches nearly every aspect of our development processes; and many things are still going to take us months (or years) to figure out
- It is a lot of work to find the most suitable 'flavor' of RE process and the right tool for your company, but deployment is even more effort (at least when coming from Word documents)
- Remember: Adoption of a requirements management tool will require the architects and engineers to change how they work, so the process and tool must help these users do their work easier, better, faster → "what's in it for me?"
  - ➢ It's this human aspect that will make or break a successful deployment → give them a voice in the process development and deployment and actively seek their feedback

- Introducing a requirements engineering process and (dedicated) requirements management tool is absolutely worth it (even though metrics can be hard to define and measure), but do not underestimate the effort and time required

Backup