

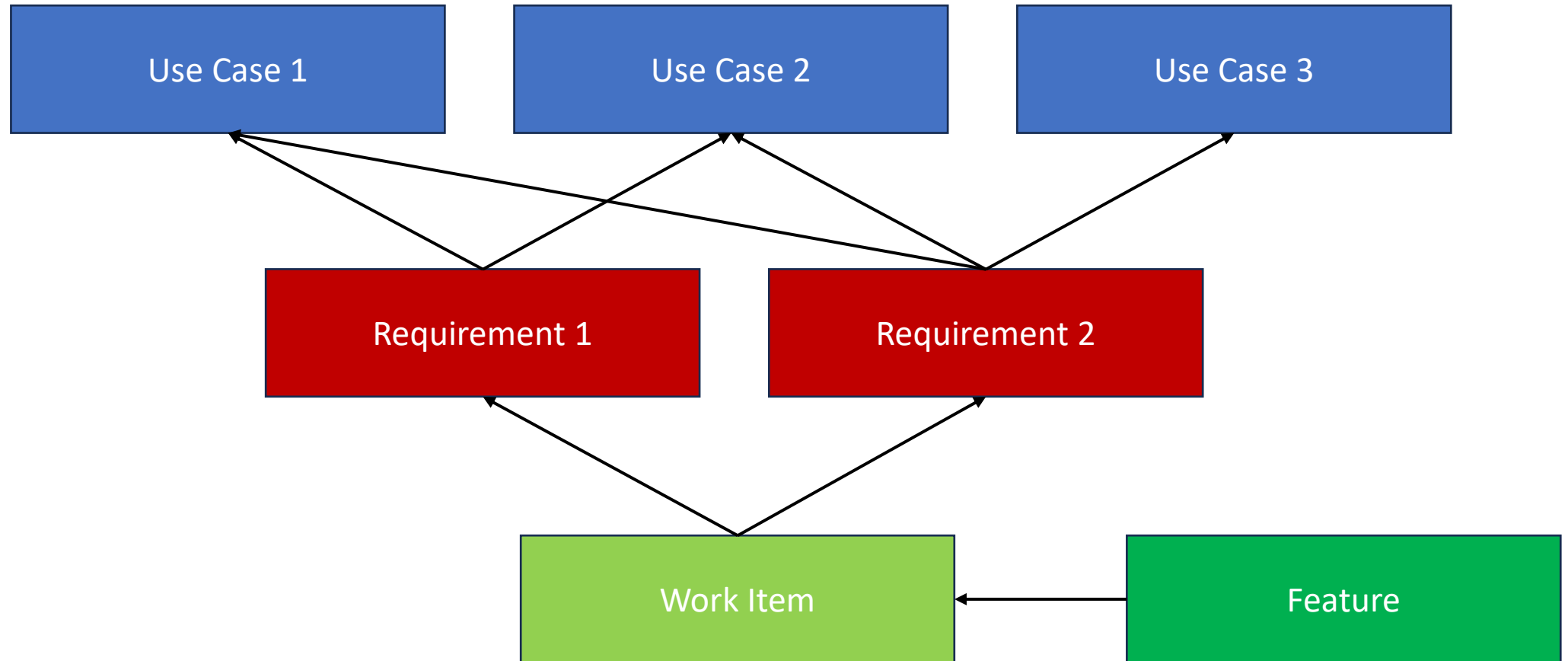
Use Cases and Requirements

Michael McCool
14 September 2023
TPAC 2023

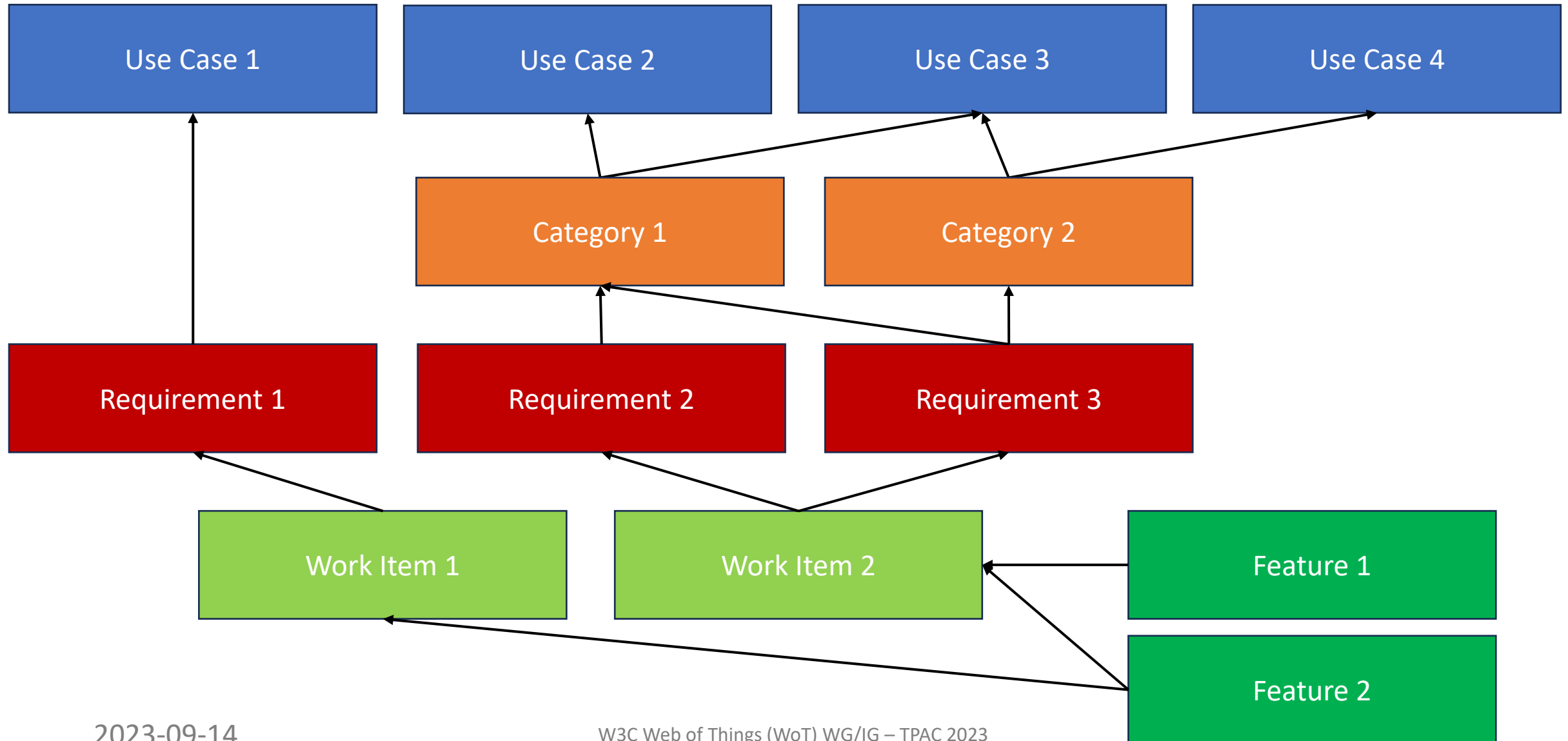
Use Cases and Requirements

- Use cases should ultimately motivate standards work:
 - Use cases should have defined requirements
 - Features should be defined to satisfy requirements
- Problems:
 - We have a large number of [new work items proposed](#)
 - but no clear connection to requirements or use cases
 - Can be difficult for authors of use cases to define all requirements
 - Existing features are also not directly tied to use cases
 - We have a large number of use cases documented but
 - these often do not specify requirements clearly
 - are not prioritized
 - have gaps and redundancies
 - Some “use cases” are really just technologies or broad domains
 - Use cases should have a defined “user”

UC → Requirement → Feature



UC → Category → Requirement → Feature



Categories

- Intermediate step but allows for generalization
 - Avoids having to constantly update “requirement” to “use case” mapping

Some possible categories:

- Private (handles personal or confidential information)
- Flexible Protocol Usage (systems use multiple protocols)
- Cloud Integration (needs to share data with remote servers)
- Local Access (needs to operate without a global connection)
- Mobile (location is subject to change)
- Resiliency (needs to be robust to failures and attacks of various kinds)

Security

- Security/Privacy Features are generally to mitigate “risks”
 - S&P sections generally each have a defined risk
 - ... then mitigations, some of which may be normative
- Risks are documented in the “Security and Privacy Guidelines” document
 - Although it needs to be updated
 - Done: DDoS Risk added
 - Still needs to be made consistent with other documents
 - E.g. Stakeholders
- ***Need to identify which use cases have which risks***

Discovery

- Original “Simple” Requirements Document can be updated
 - <https://github.com/w3c/wot-usecases/blob/main/REQUIREMENTS/discovery.md>
- Probably also want to connect to geolocation requirements
- Can inline into Use Cases and Requirements Document.
 - Expand existing 1-paragraph “Discovery Requirements” section
- Note:
 - Current Discovery spec does not meet the requirements stated in either location!
 - ***Notably, “queries” are cited as a requirement in Architecture 1.1 but while some mechanisms are defined, they are not mandatory in directories***

Suggested Plan

- Expand “Requirements” Section in Use Cases and Requirements document to define requirements (if not defined elsewhere) and connect them to use cases.
 - Want to avoid editing use cases themselves for authorship and consolidation reasons
- Keep it simple:
 - A named requirement and a few sentences defining each one is enough.
 - If defined elsewhere (e.g. security risk) can link to another document for definition.
 - However, best IMO if requirements are in a “published” document, not a random MD file somewhere...
 - Links to use cases motivating each requirement
 - MAY use intermediate categories...
 - Can have “general” and “sub” requirements
 - E.g. Support interoperability → support multiple protocols → support flexible binding mechanism

Discussion

Sb: how should we differentiate use cases for which purpose, esp for “small features” – do really need a use case?

Mm: Features motivated by requirements and can have general requirements like “ease of use” for smaller things

Kaz: Definitely need to base spec work on use cases. Mechanisms which make things “easier” should still be motivated by use cases. Technical discussions should be based on use cases.

Mm: also worth saying: Easier for who? Use cases capture that.

Ege: Want to avoid being too generic. Some requirements are derived from “mission statement”. Want to avoid strange interactions. May also be useful to divide work items between new features and redesign/reorg/policy issues.

Kaz: MJK’s proposal can be discussed more in TD call. Basically agree with MM. Can start with high-level (“vague”) requirements first. Need to think more about how to structure this some more.

MM: Suggest doing a draft based on proposals above for Discovery then see how it works, then discuss more.

Discussion (continued, from TPAC 2023)

- Singleton categories – only one use case – do we need a category?
- Sub-requirements
- “Satisfiability” – large requirements may be “ongoing” in nature (e.g. “ease of use”), but can define “satisfiable” sub-requirements.
- May be multiple features per work item, etc.
- Need clear identifiers for use cases, requirements, etc. (e.g. UC-SA, R1, etc).
- Tooling to keep track of cross-references?
 - E.g. “tag” use cases with requirements/categories, then generate links?
 - Daniel mentioned there WAS some tooling early on for this

Links and Resources

- Use Cases and Requirements Document
 - <https://w3c.github.io/wot-usecases/> (Editor's draft, rendered)
- Requirements Template
 - <https://github.com/w3c/wot-usecases/blob/main/REQUIREMENTS/requirements-template.md>
 - Example: Geolocation
 - <https://github.com/w3c/wot-usecases/blob/main/REQUIREMENTS/geolocation-requirements.md>
- Requirements Summary
 - <https://github.com/w3c/wot-usecases/blob/main/REQUIREMENTS/requirements-summary.csv>
- Older Requirements Document (Discovery):
 - <https://github.com/w3c/wot-usecases/blob/main/REQUIREMENTS/discovery.md>