|  |
| --- |
| PTC |
| Multi Context Agreements |
| Approach |

|  |
| --- |
| V V L Sudha Kancharla  5/24/2017 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Approach | | | | |
| Release Theme | [RT 626951](http://integrity.ptc.com:7001/im/issues?selection=626951) |  | | |
| Planned Release | Eagle | Owning Team | | Security WPD |
| T-Shirt Estimate | Small (60 days) | Approach Owner | | Pritam Ghiya / Chintan Anand |
| Status | Draft | Product Owner | | Laura Nissen |
| Review Team | | | | |
| Scrum Master |  | |  | |
| Technical Lead | Pritam Ghiya | |  | |
| Architecture Rep |  | |  | |
| Product Owner | Laura Nissen | |  | |
| UX Representative |  | |  | |
| QA Representative | Snehal Dange | |  | |
| PUBS Representative | Manank Gor | |  | |

|  |  |  |  |
| --- | --- | --- | --- |
| # | Reviewer | Status | Issue/Resolution |
| 1 | <Reviewer Name> | Open | Issue: <Description of the Issue>  Resolution: <Response to the Issue> |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |
| 11 |  |  |  |
| 12 |  |  |  |

*For complete details on the content of the approach, refer to the “*[*Define Approach*](http://rdwiki.ptcnet.ptc.com/mediawiki/index.php?title=DefineApproach)*” process located on the process website. The approach process includes an “*[*Approach Readiness*](https://pds.ptc.com/Windchill/app/#ptc1/tcomp/infoPage?ContainerOid=OR%3Awt.projmgmt.admin.Project2%3A821398934&oid=VR%3Awt.doc.WTDocument%3A907184739&u8=1)*” checklist that can be used to guide the content.*

# Strategy

## Background and requirements.

Today the Context Based Agreements can be created on any context, For example on a product context or project or organization. The Agreement will be applied to all the objects inside the container, where it is created.

It is challenging, for the customers who are having 1000s of contexts (Either projects or products or libraries etc.,) where similar kind of agreement is to be created at multiple contexts. In this case creation, maintaining of numerous similar agreements is also very hard.

The goal of this Release theme is to provide a provision to create a Multi Context Based Agreement objects for the user, where a single agreement instance can be applied to multiple contexts.

This new feature should take care of all below listed functionalities:

* Agreement UI Operations – Create, Edit, View
* Related Agreements View
* Access Permission check API (has Access)
* Update Cache

## Existing behavior:

* Context Based Agreement points to single context. It applies to all the objects present in that context.
* The context information is not explicitly stored in database. The context where this agreement gets created is considered as the authorized context for context based agreement.
* Access permission check looks only for the container of the agreement and compares that with the object container.

## Proposed solution:

* Context Based Agreement will hold multiple contexts. It will be applicable to all the objects of the selected contexts.
* The Authorized contexts will be persisted in DB as Authorized Agreement Objects.
* Access permission check will look for the container of the agreement for single context agreement and authorized contexts for multi context agreements. These both type of agreement will coexist. Hence there is no need of any migrator.
* While creating context based agreement, search result will show all sub contexts of current context. User will have to pick up all applicable contexts. Current context will be preselected in applicable contexts table. *This is currently not decided whether the pre-selected checkbox will be read-only or could be editable.*
* Info page of context based agreement will list contexts in new tab called “Authorized Contexts”.

1. **Agreement UI Operations:**

1. Create Agreement

In the New Agreement window, the subtabs vary depending on the type of agreement.

For the Context Based Agreement, there should be a new tab for selecting authorized contexts with a context picker. This window should display all sub contexts for the user to multi select. In the context picker current context should be pre-selected. These Selected context objects should be saved in DB as Authorized Objects.

1. Edit Agreement

The Edit Agreement Window will allow editing of the selected contexts. The editing of any existing single context agreement should convert to multi-context agreement. Respective cache should also get updated.

1. View Agreement

In current implementation for Context based agreement, it has two tabs: Details & Authorized Participant. We will add a new Tab “Authorized Contexts”- For Context Based Agreements. In case of existing single context agreement, it will list the current context, whereas for multi-context agreement all the respective context will be listing in this table.

1. **Related Agreements View:**

For the object, its context will be fetched and the corresponding multi and single context agreements will be shown in Related Agreements table on the object information page.

1. **Access Permission check API:**

Access permission check will look for the container of the agreement for single context agreement and authorized contexts for multi context agreements.

1. **Cache Update:**

There will be change in the existing cache mechanism but no impact to cache size.

The way we store the meta data in Tier-3 cache will be changed to accommodate multi-context references as well. A List variable to store the respective contextReferences will be added in AgreementMetaDataCachecAuthVersionRange.java which will be used while storing multiple contextReferences in cache. During get call, instead of referring getContainerOid from AgreementMetaDataCacheEntry.java, the list from AgreementMetaDataCachecAuthVersionRange.java will be referred to get all the contextList associated with the Agreement.

1. **Components:**

The main components/work tasks for this approach are:

* Modify Create Agreement UI handler in order to provide below features:
  1. Modify the tabs, On Selecting the ContextBased Agreement add the selected context tab.
* Modify the Edit Agreement UI handler to include the tab which helps to edit the contexts corresponding to agreement.
* Modify the View Agreement Window, to show the authorized contexts of ContextBased agreement.
* Modify the api which fetches the related agreements for Related Agreements view, in order to include the Multi Context Based Agreements in this view.
* Implement the changes at hasAuthorization check of agreements, to consider Multi Context Based Agreements also, while checking against the agreement metadata.

# Assumptions

No performance impact. As there is no database change because we are leveraging the existing table, hence no migrator is needed.

# Issues and Risks

We are not anticipating any risk for single context agreement to be treated as multi-context. The UI will be consistent for the old single context agreement in the new implementation showing the context in Authorized Context (new) tab.

# Dependencies

None

# Alternative Approaches

## Multi context agreement type

* As an alternative approach, we can create Multi-context based agreement type, and then have authorized contexts for this agreements.
* Agreements will be persisted in DB as authorized agreements, and the related contexts as authorized contexts in DB.
* UI Changes, Access Permission Check changes and Related Agreements changes listed for actual approach, remains the same.
* Requires changes in the Agreement Configuration and all dependent changes.
* No Migrator impact.

# References

|  |  |
| --- | --- |
| Name | Comments |
|  |  |
|  |  |
|  |  |