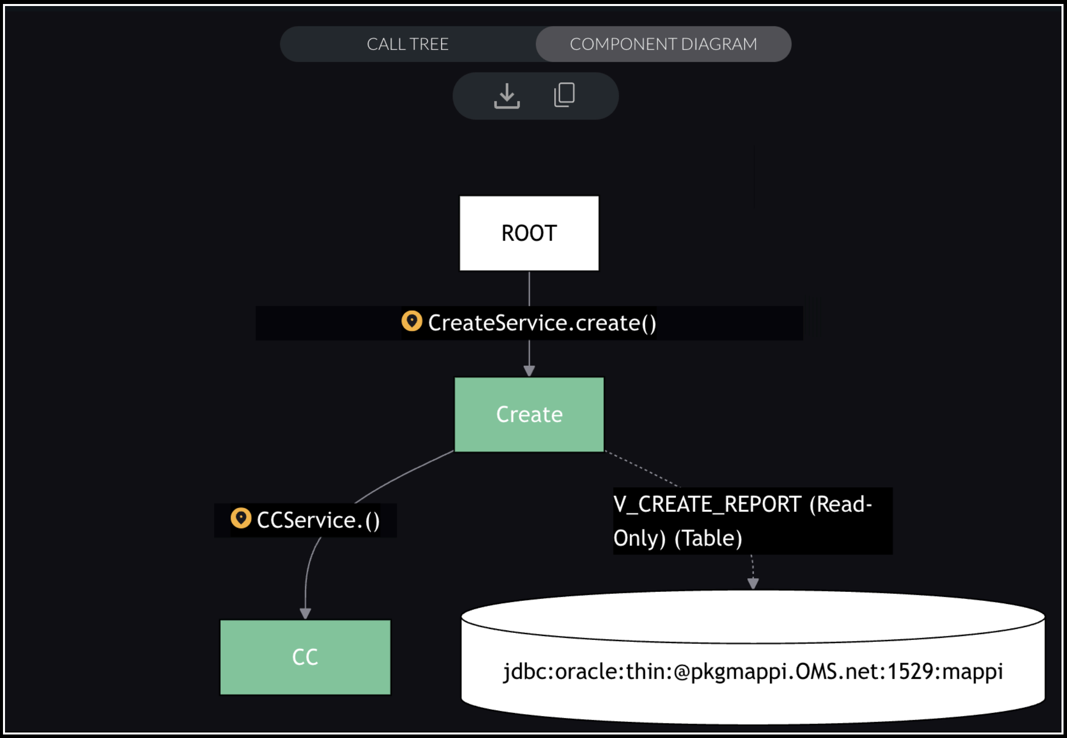
vFunction 4.2 Release Notes

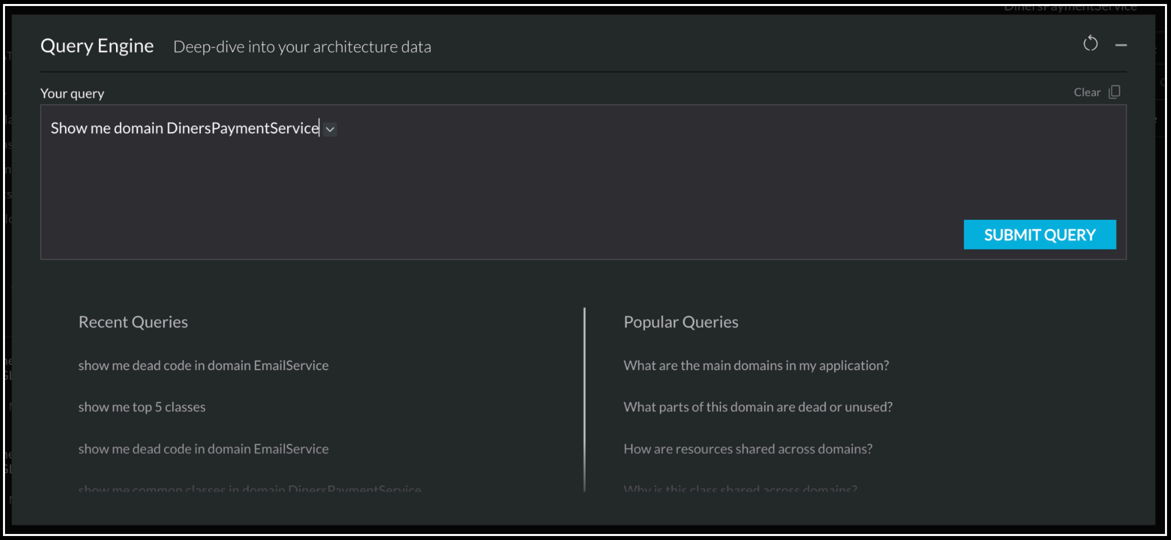
Key Additions

- **Generate C4 Component Diagrams for Monolithic applications**
  vFunction now supports generating and exporting C4 component diagrams from the call tree. The diagrams take into account architectural boundaries, calls between domains and access to resources, making it easier to understand system structure and highlight architecture issues across domain boundaries.


CALL TREE
COMPONENT DIAGRAM
ROOT
CreateService.create()
Create
CCService.()
V_CREATE_REPORT (Read-Only) (Table)
CC
jdbc:oracle:thin:@pkgmappi.OMS.net:1529:mappi

- **GenAI Powered Query Engine**
  This feature bridges the gap between natural language and technical data. By translating user written prompts into internal vFunction DB queries, the Query Engine opens new ways for users to explore and retrieve information based on their specific needs. This offers a new level of accessibility, empowering teams to uncover application specific insights for their app requirements.
  These queries can be for the entire measurement or contextual, either from a specific node in the Call Tree, or related to a specific domain.

vFunction

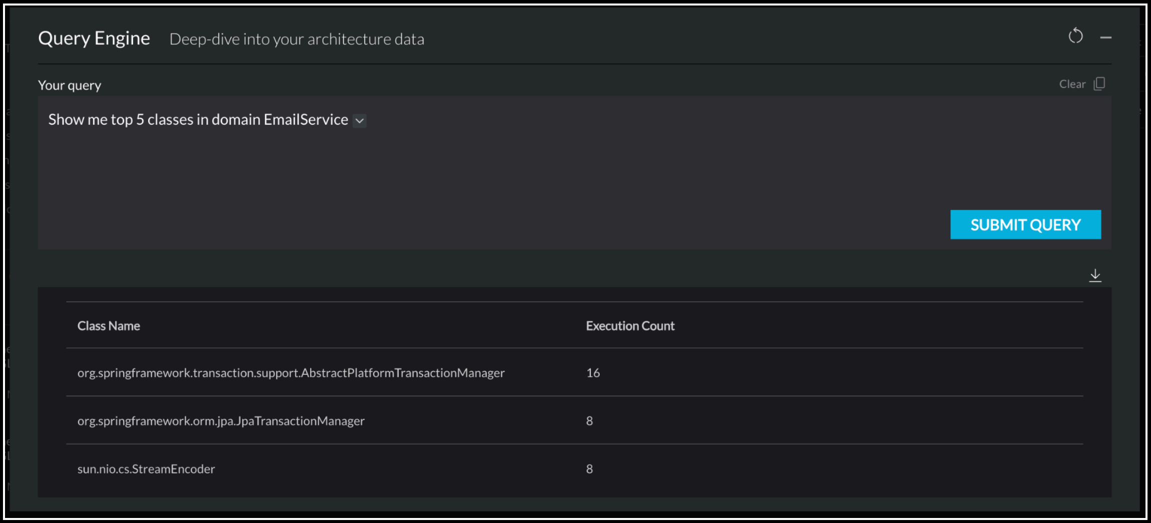

Query Engine Deep-dive into your architecture data

Your query

Show me domain DinersPaymentService

Recent Queries	Popular Queries
show me dead code in domain EmailService	What are the main domains in my application?
show me top 5 classes	What parts of this domain are dead or unused?
show me dead code in domain EmailService	How are resources shared across domains?

Once the query is executed successfully, users are presented with the results, along with an option to download the data as a CSV.


Query Engine Deep-dive into your architecture data

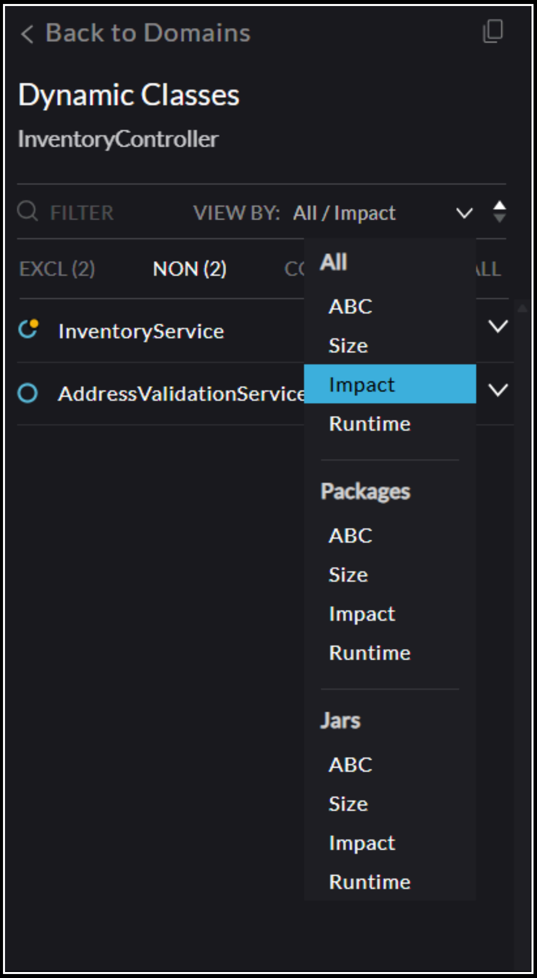
Your query

Show me top 5 classes in domain EmailService

Class Name	Execution Count
org.springframework.transaction.support.AbstractPlatformTransactionManager	16
org.springframework.orm.jpa.JpaTransactionManager	8
sun.nio.cs.StreamEncoder	8

Important note: Only the query is sent to the GenAI provider. The result of the query is in a form that the vFunction database can process. Only then is the query executed on the data stored within the vFunction server and presented to the user.

- **Sorting classes by total runtime and their impact on modularization**
  Classes in each domain can now be sorted based on their runtime percentage, to highlight performance-critical components. Non-exclusive classes can now be sorted based on their impact on the application's complexity. The higher the impact the more refactoring these classes will contribute to the overall modularity.



< Back to Domains

Dynamic Classes

InventoryController

FILTER VIEW BY: All / Impact

EXCL (2) NON (2)

InventoryService

AddressValidationService

Usability Improvements

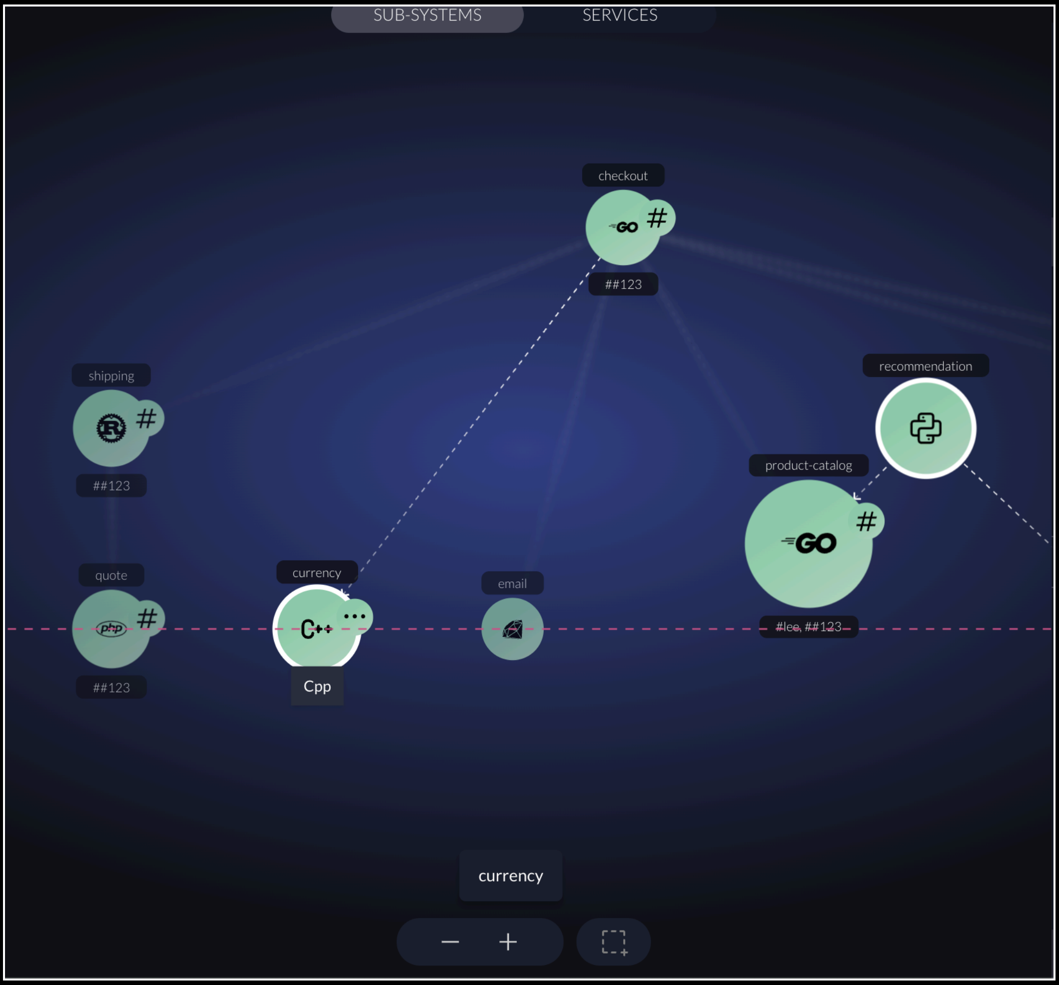
- **Controller Selection by Name and Tag**
  Instance ID selection has been removed to avoid ambiguity in dynamic environments. Manual learning configuration now supports selecting controllers by name or tag, instead of instance IDs. Users can choose to learn from a single controller or all controllers matching selected names or tags. Viper controllers follow the same selection logic when Refresh Viper is enabled.

- **Recalculate TODOs**
  Users can now manually trigger a "Recalculate TODOs" action from the Expanded TODO List's new Actions menu next to the TODO and DONE counts in the upper-right corner. This regenerates the list of automatic TODOs by comparing the latest measurement with the current baseline:

- ○ All automatic TODOs (including resolved and dismissed) are deleted and recreated.
- ○ Manual TODOs are preserved, with their timestamps updated.
- ○ JIRA integration posts a comment on any linked tickets indicating the TODO was deleted.

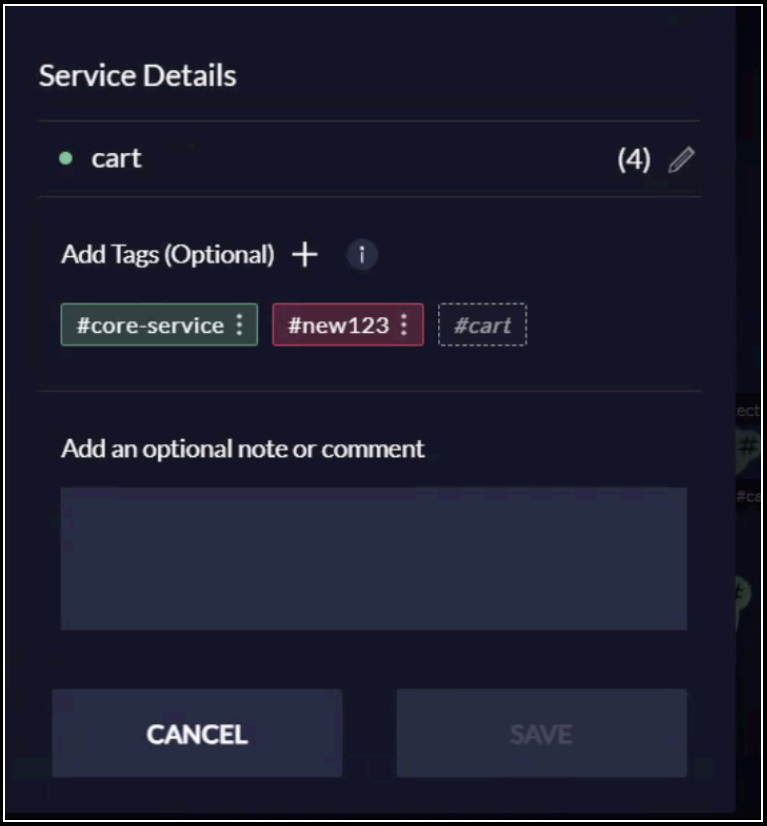
New Capabilities for Distributed Applications supported by OTEL

- **Manually setting service architectural layer**
  This enhancement gives architects greater control over layering decisions.
  Architects can now manually assign services to specific architectural layers, overriding the AI-calculated layer assignments.
  Once a service is moved, TODOs are automatically recalculated based on the updated service layering, allowing teams to reflect intentional architectural decisions directly in the platform.



Note: This capability works in the sub-systems graph, and not in the services graph, since internal sub-system layering is omitted from the todo calculation of the parent application.

- Service Tag Comparison Between Measurements and their References
  Added tag comparison between the latest or baseline measurement and the latest reference, enabling validation of user defined, reference defined and attribute based service tags for consistency and completeness:
  - Green – Tag exists in both measurement and reference.
  - Red (solid border) – Tag exists only in measurement or only in reference (user/ref defined).
  - Red (dashed + italic) – Tag is attribute-based and exists only in measurement.



Service Details

● cart (4)

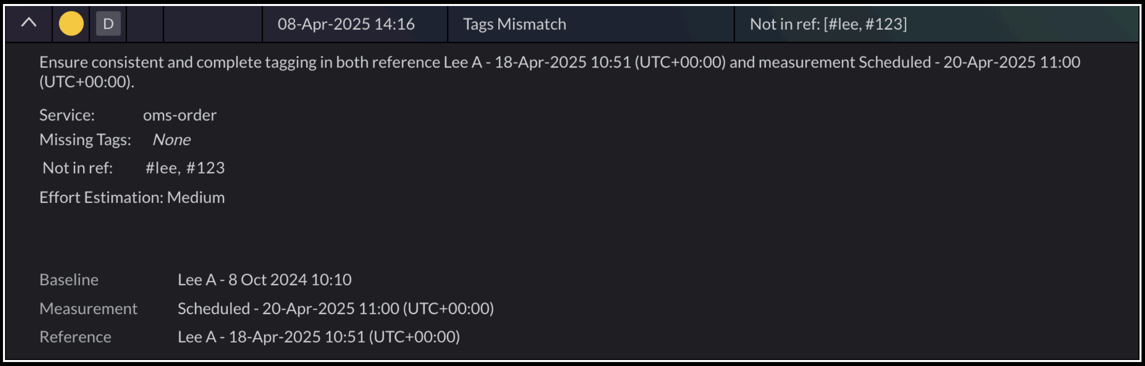
Add Tags (Optional) +

#core-service #new123 *#cart*

Add an optional note or comment

CANCEL SAVE

- New TODO for distributed applications - Tag Mismatch
  A new Tags Mismatch TODO highlights services that are missing expected tags or have inconsistencies when compared to the latest reference. This helps ensure consistent tagging across services, helps in validating deployment architecture compared to the approved architecture diagram, validates meta-data and more.



08-Apr-2025 14:16 | Tags Mismatch | Not in ref: [#lee, #123]

Ensure consistent and complete tagging in both reference Lee A - 18-Apr-2025 10:51 (UTC+00:00) and measurement Scheduled - 20-Apr-2025 11:00 (UTC+00:00).

Service: oms-order
Missing Tags: *None*
Not in ref: #lee, #123
Effort Estimation: Medium

Baseline	Lee A - 8 Oct 2024 10:10
Measurement	Scheduled - 20-Apr-2025 11:00 (UTC+00:00)
Reference	Lee A - 18-Apr-2025 10:51 (UTC+00:00)

- **Preventing Changes to Groups and Tags on the Latest Measurement**
  To preserve analysis accuracy, users can no longer modify groups, group tags and service tags on the latest measurement.