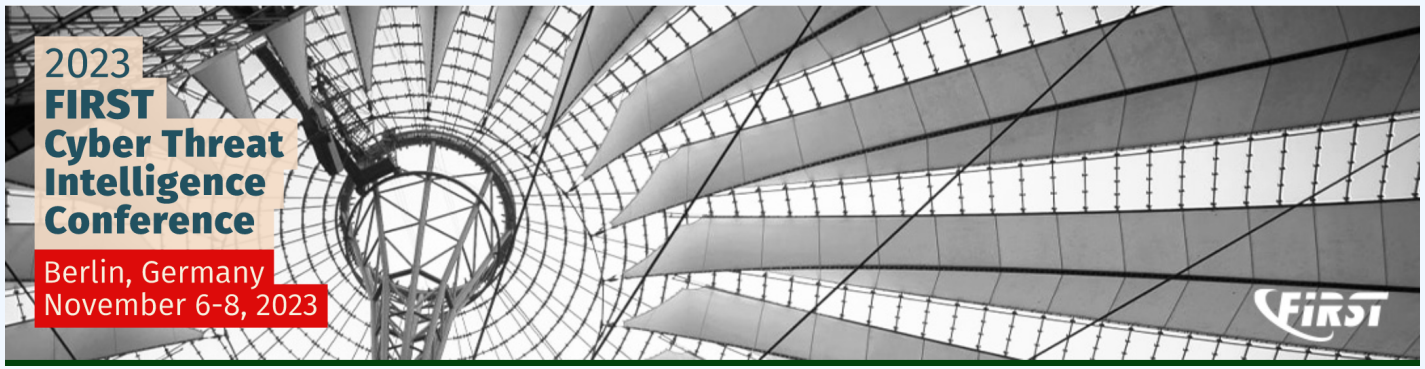
MISP 3 - Teaching an Old Dog New Tricks

Paving the way forward

Andras Iklody & Sami Mokaddem

MISP Project
https://www.misp-project.org/



$ whoarewe

Andras Iklody
iglocska

@iglocska

Sami Mokaddem
mokaddem

@mokaddem_sami

Agenda

- Why MISP 3?
- The plan
- Considerations

Why MISP 3?

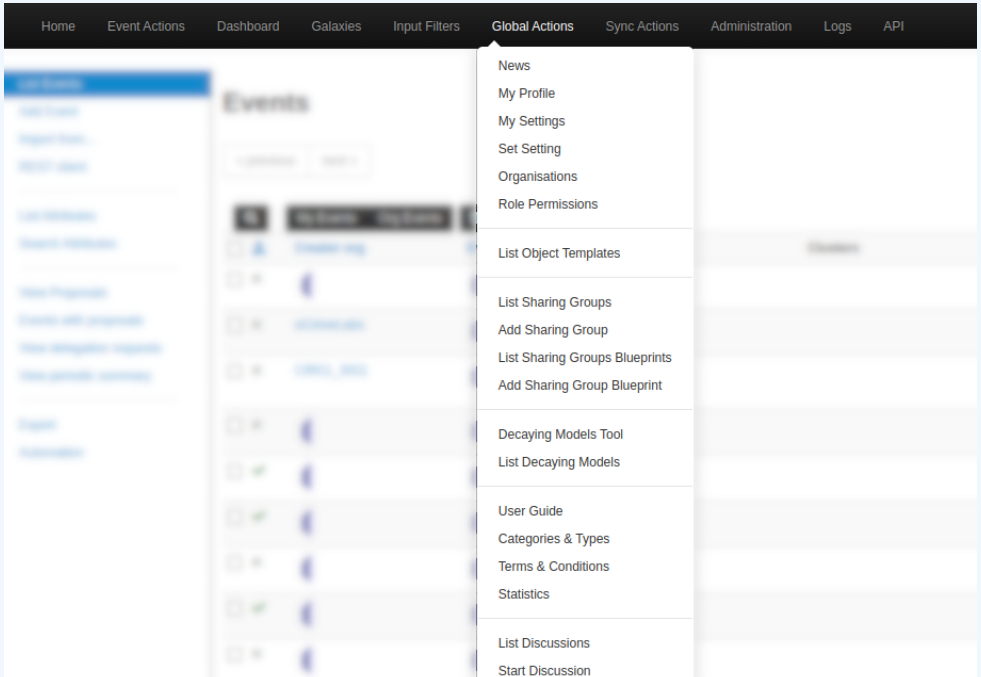
An outdated version of the framework

CakePHP
Build fast, grow solid.

- MISP is based on CakePHP 2.x
  - End of Security Support in **June 2021**
  - Maintained fork github.com:MISP/cakephp.git
- CakePHP supports PHP version **<=7.4**
  - End of Security Support in **November 2022**

Tacked on mechanics

- MISP supports a wide range of use cases...
- ... meaning loads of feature-clutter the interface
- All options visible regardless of the user profile
- Lack of coherent page navigation



Shortcomings due to initial design choices

To list a few..

- Sub-optimal database structure
- Start with something small, build it out has its disadvantages
  - Attribute `type`, `value` not a first-class citizen
  - Logs all in one place
  - Indexing rework (performance and moving validation to the DB)
- Confusing mess of multiple graphing interfaces
- Files - Especially tricky with dockerised and load balanced setups
- Tagging

The ongoing plan forward

- Port of the codebase to a new stack
  - CakePHP 2.x → CakePHP 5
- Rework of old baggage
  - Database updates
  - Front-end libraries (Bootstrap, Graphing, …)
  - Background jobs & Scheduled tasks
  - Purging old libraries

Pruning unused / dead end functionalities

- Populate using the templating system
- Deprecated export functionalities
- Discussion / Posts
- …

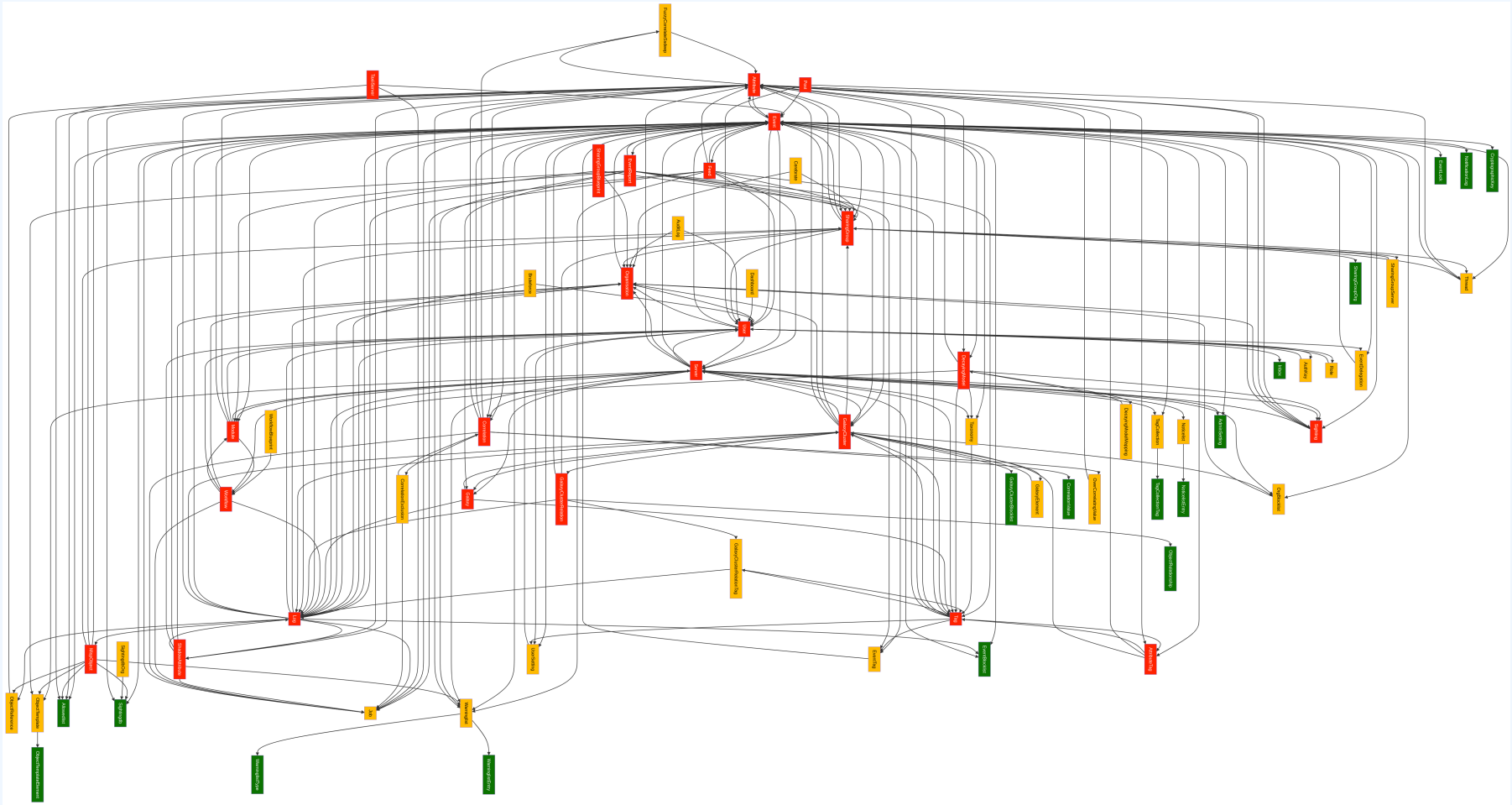
Step I - Preparing the grounds

Step I - Preparing the grounds

- Refactoring the codebase for improved portability using factories
  - Framework-agnostic
  - Reusable code for front and back-end
  - Extracting and encapsulating specialised functionalities into libraries
- Setting the stage with Cerebrate
  - Dev started in May 2020, built on MISP3's stack
  - Application built on top of ported MISP libraries
  - New UI laying the foundation for MISP 3
  - Streamlined integration of new features into MISP3
    - Tagging, Inbox system, Settings, ⋯

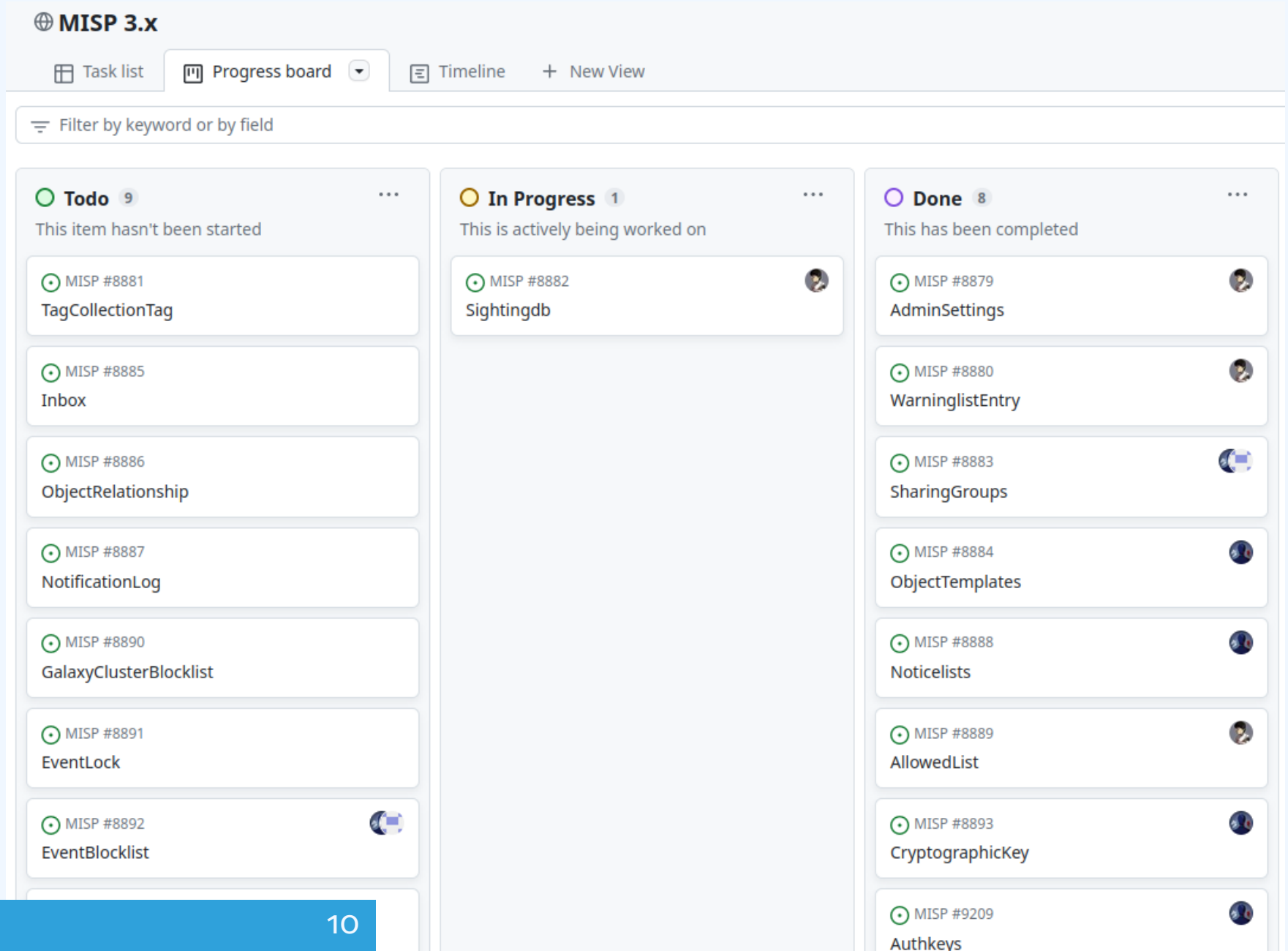
Step I - Identifying inter-dependencies

Migrate least connected part first



Step II - Porting the codebase

Step II - Roadmap for a 3-wave porting



Step II - Roadmap for a 3-wave porting

Wave 1 Least complex/inter-connected models

- E.g. Blocklist, Warninglist, Object-template, User

Wave 2 More glue relying on component already migrated

- E.g. Authkey, *-Tag, Taxonomy

Wave 3 The actual meat of the application

- E.g. Attribute, Event, Workflow

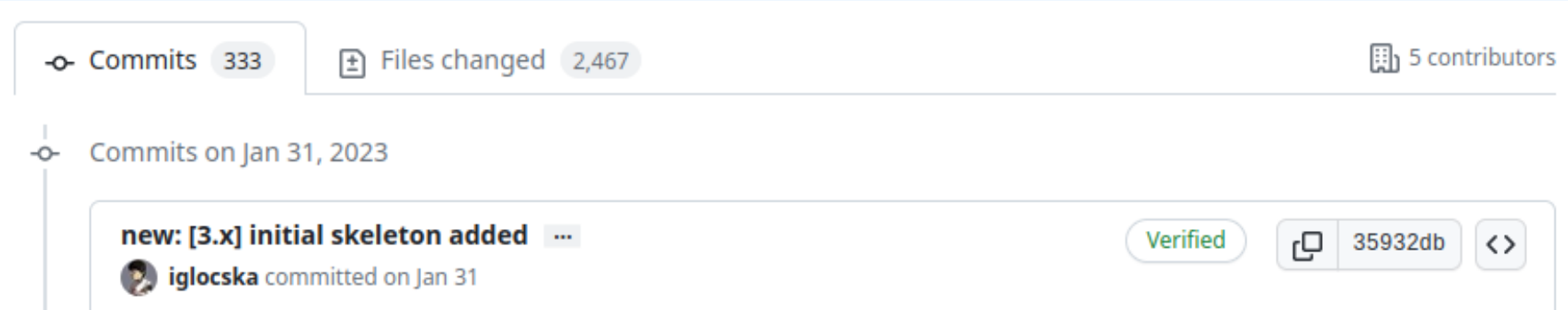
Step II - Test driven development

```
$ composer test
> sh ./tests/Helper/wiremock/start.sh
WireMock 1 started on port 8080
> phpunit
[ * ] Running DB migrations, it may take some time ...

The WireMock server is started .....
port:                         8080
enable-browser-proxying:      false
disable-banner:               true
no-request-journal:           false
verbose:                      false

PHPUnit 8.5.22 by Sebastian Bergmann and contributors.
```

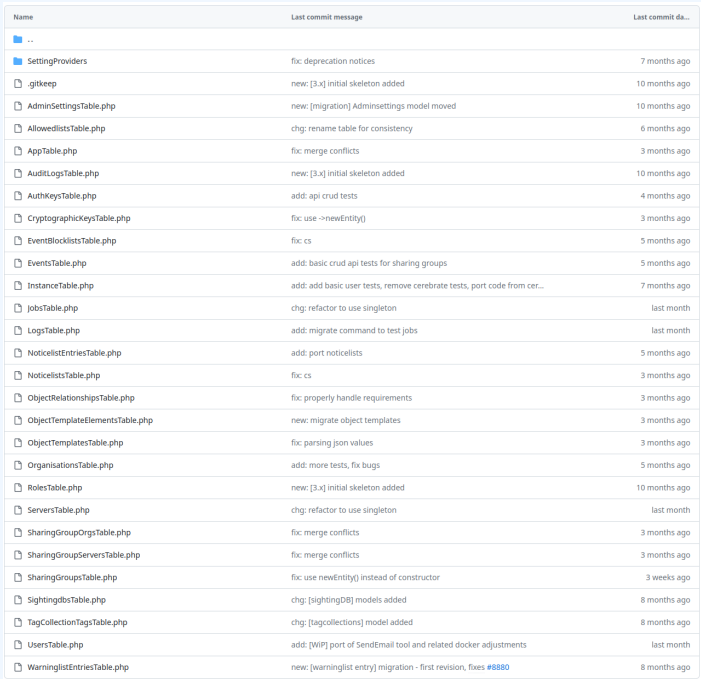
- Complementary to PyMISP test
- In-framework **Unit Tests** and **Endpoint Tests**
- Improved CI pipeline and enforced code standard

Codebase Migration: Where We Stand I

Migration officially started in January 2023

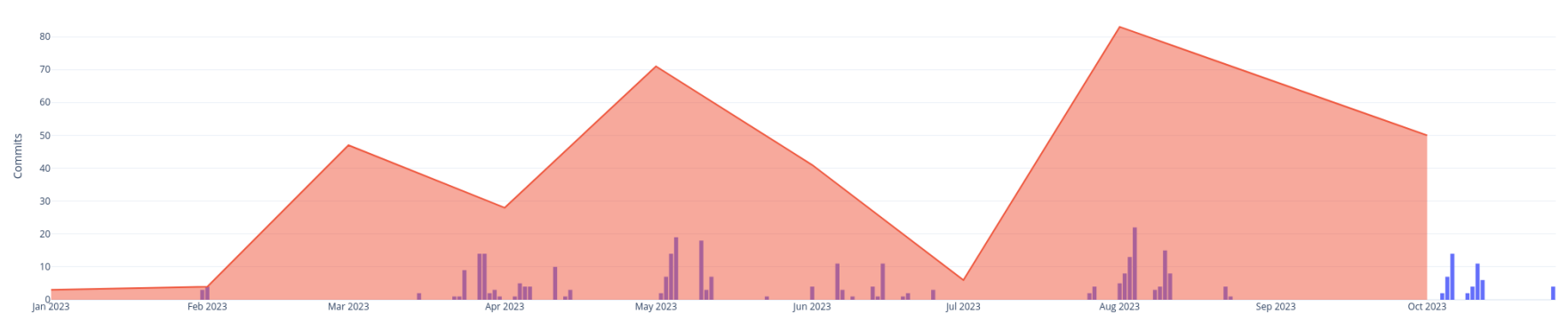


- Around **27 tables** have been moved
- Some partially, others completely

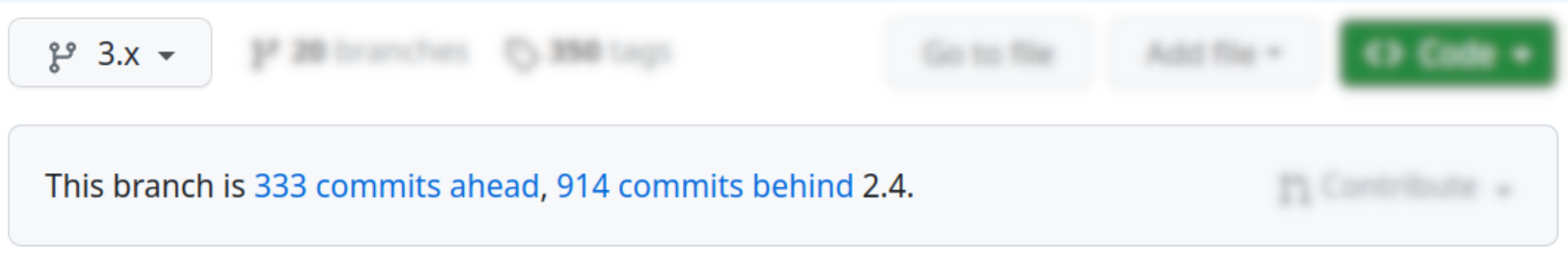


Name	Last commit message	Last commit da...
..		
SettingProviders	fix: deprecation notices	7 months ago
.gitkeep	new: [3.x] initial skeleton added	10 months ago
AdminSettingsTable.php	new: [migration] Adminsettings model moved	10 months ago
AllowedlistsTable.php	chg: rename table for consistency	6 months ago
AppTable.php	fix: merge conflicts	3 months ago
AuditLogsTable.php	new: [3.x] initial skeleton added	10 months ago
AuthKeysTable.php	add: api crud tests	4 months ago
CryptographicKeysTable.php	fix: use ->newEntity()	3 months ago
EventBlocklistsTable.php	fix: cs	5 months ago
EventsTable.php	add: basic crud api tests for sharing groups	5 months ago
InstanceTable.php	add: add basic user tests, remove cerebrate tests, port code from cer...	7 months ago
JobsTable.php	chg: refactor to use singleton	last month
LogsTable.php	add: migrate command to test jobs	last month
NoticelistEntriesTable.php	add: port noticelists	5 months ago
NoticelistsTable.php	fix: cs	3 months ago
ObjectRelationshipsTable.php	fix: properly handle requirements	3 months ago
ObjectTemplateElementsTable.php	new: migrate object templates	3 months ago
ObjectTemplatesTable.php	fix: parsing json values	3 months ago
OrganisationsTable.php	add: more tests, fix bugs	5 months ago
RolesTable.php	new: [3.x] initial skeleton added	10 months ago
ServersTable.php	chg: refactor to use singleton	last month
SharingGroupOrgsTable.php	fix: merge conflicts	3 months ago
SharingGroupServersTable.php	fix: merge conflicts	3 months ago
SharingGroupsTable.php	fix: use newEntity() instead of constructor	3 weeks ago
SightingdbsTable.php	chg: [sightingDB] models added	8 months ago
TagCollectionTagsTable.php	chg: [tagcollections] model added	8 months ago
UsersTable.php	add: [WiP] port of SendEmail tool and related docker adjustments	last month
WarninglistEntriesTable.php	new: [warninglist entry] migration - first revision, fixes #8880	8 months ago

Codebase Migration: Where We Stand II



- Migration speed ramping up. The more we port, the faster we go

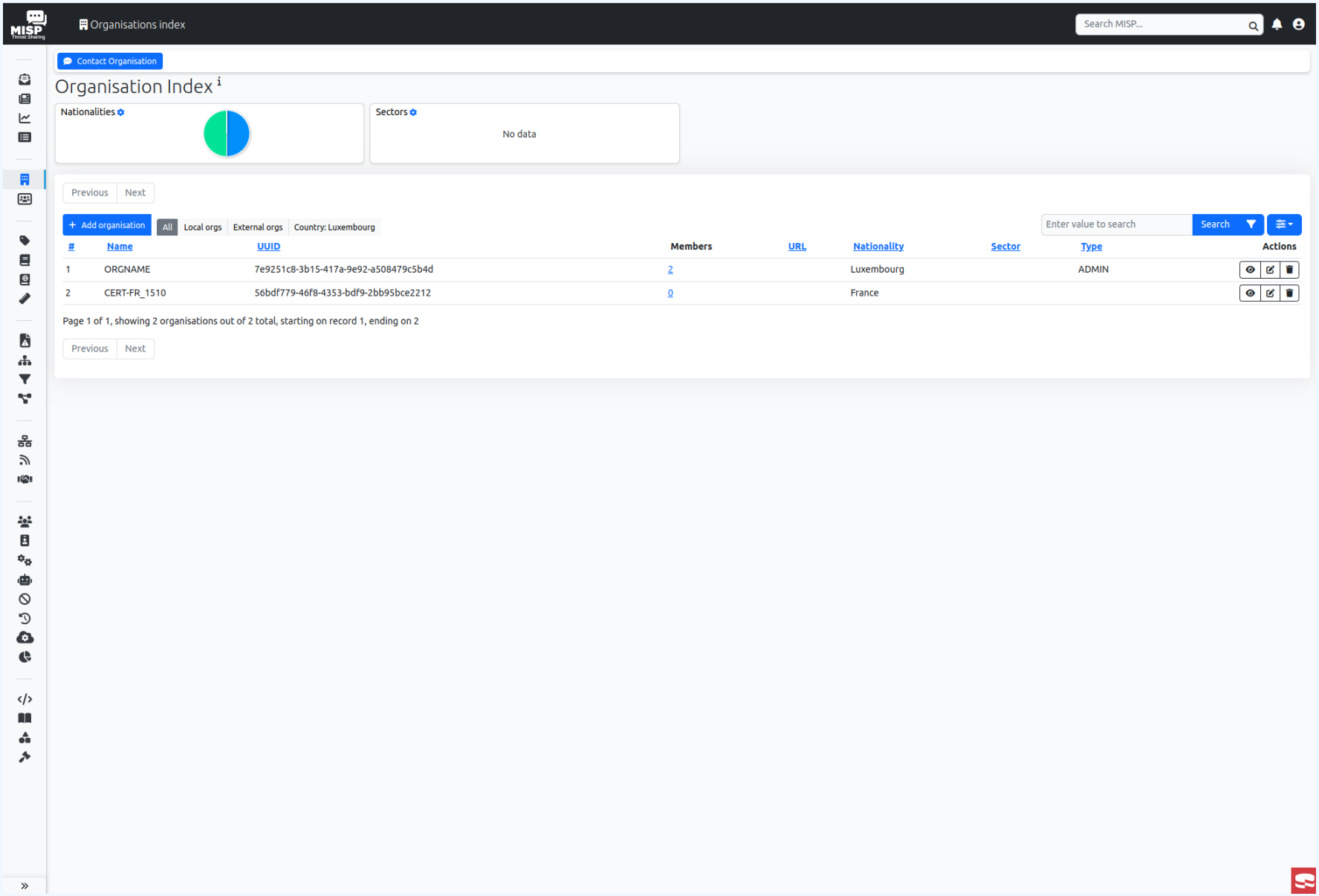


- Even while supporting and improving `2.4`

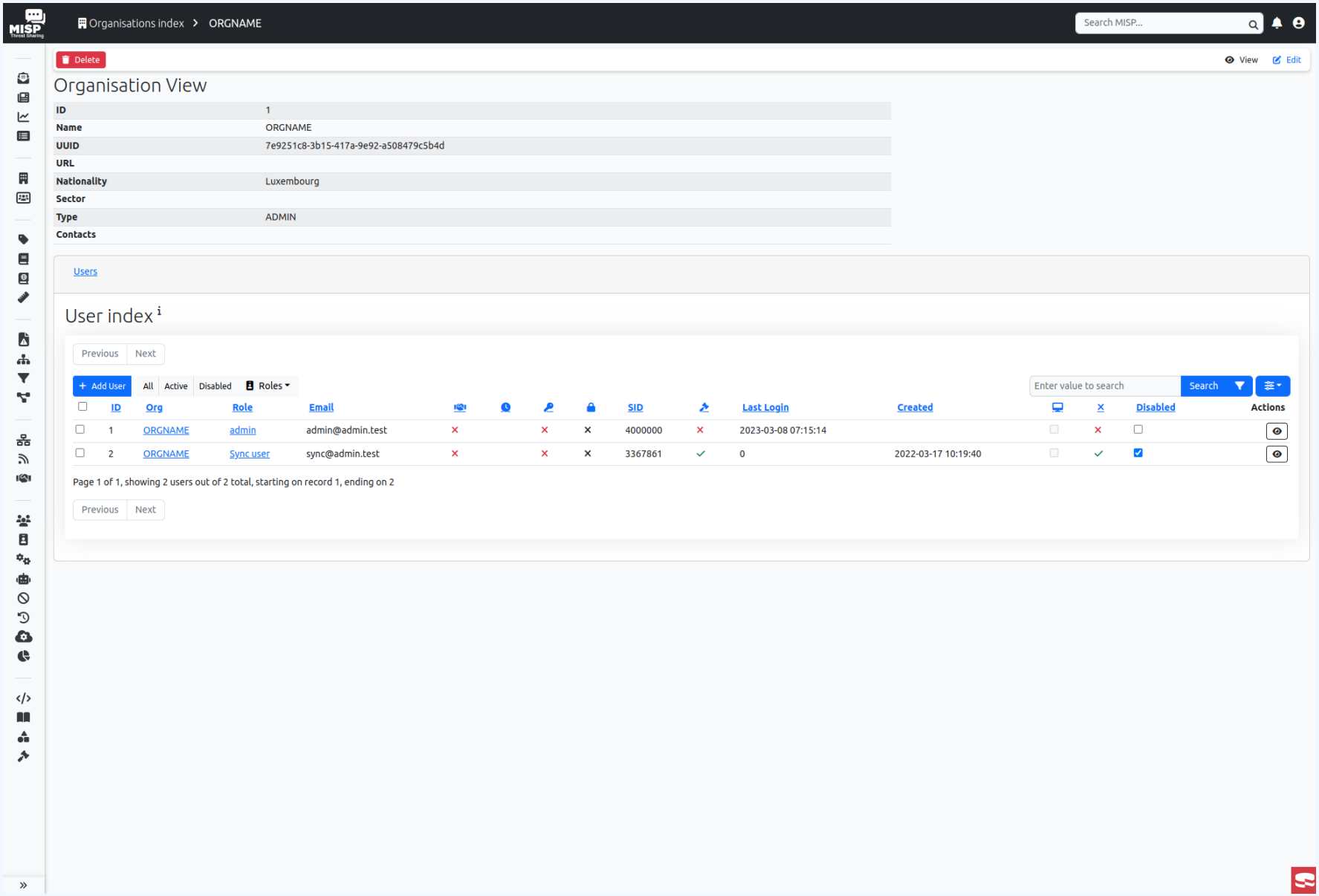
Codebase Migration: Look and Feel I

- Most of the changes are **invisible**
- Some user interfaces can still be displayed

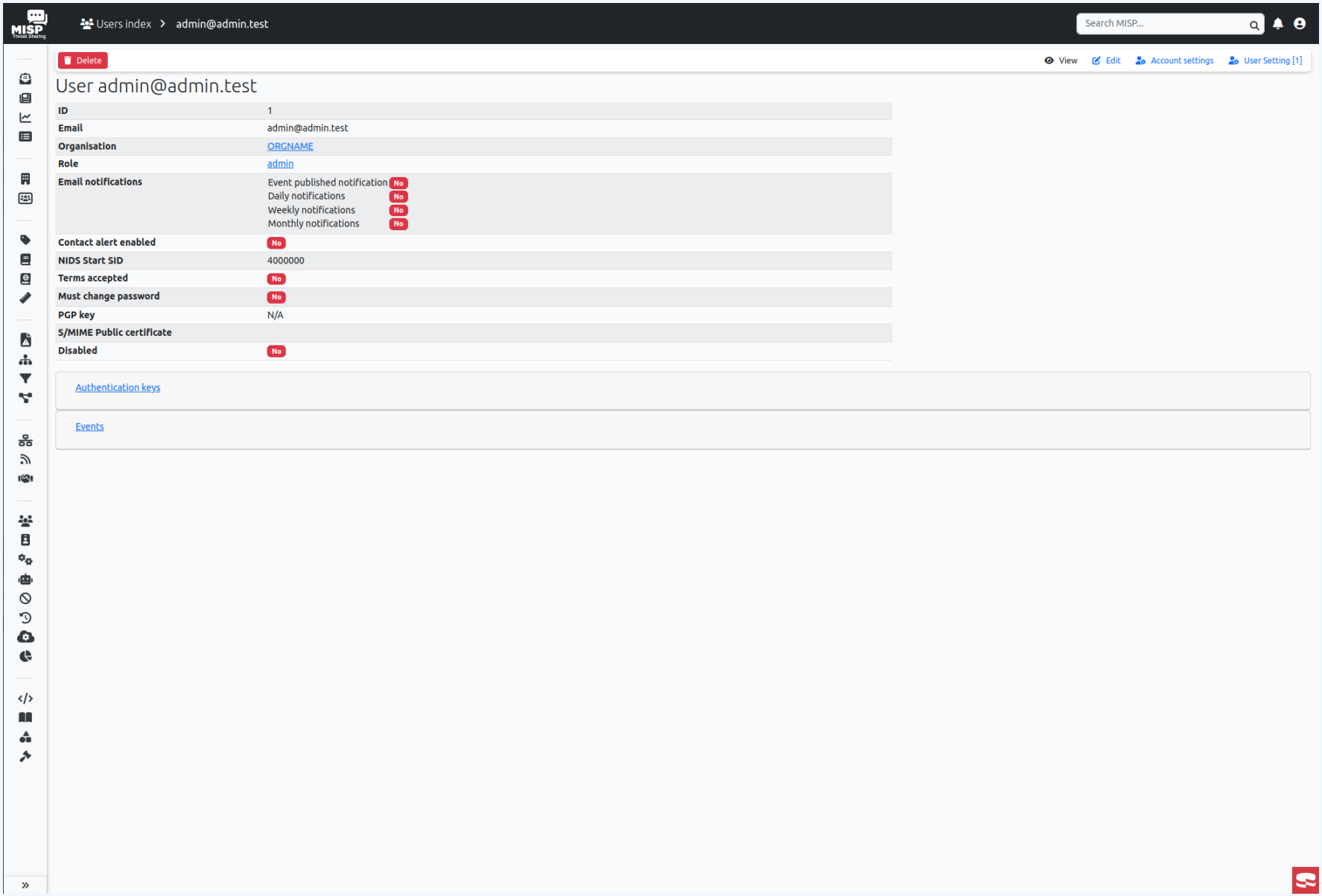
Codebase Migration: Look and Feel II



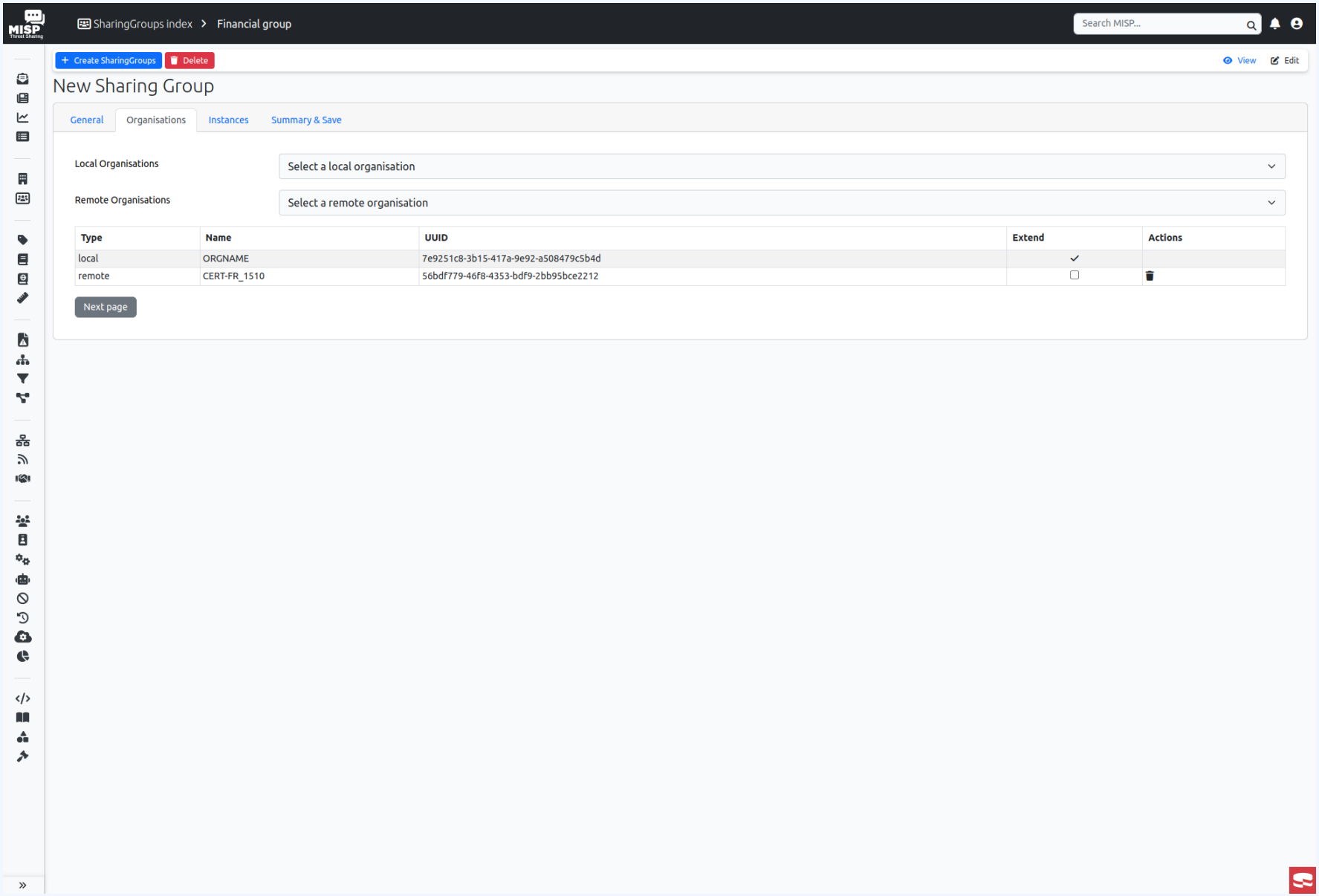
Codebase Migration: Look and Feel II



Codebase Migration: Look and Feel II



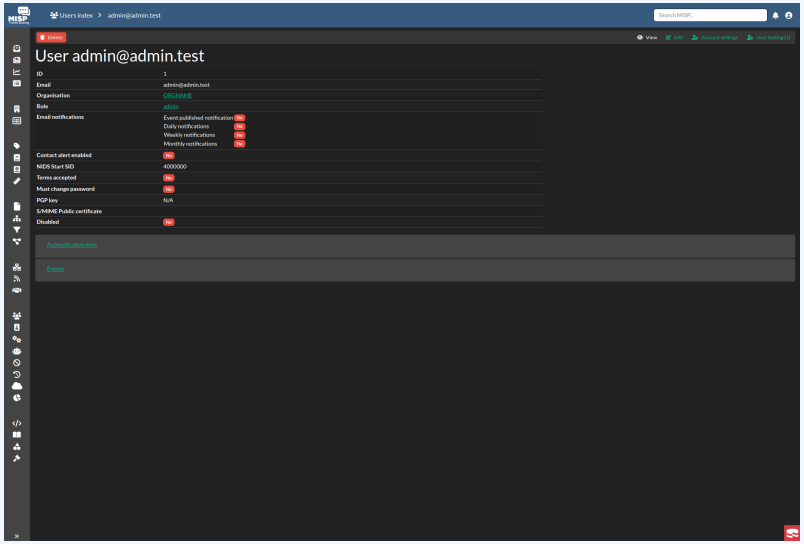
Codebase Migration: Look and Feel II

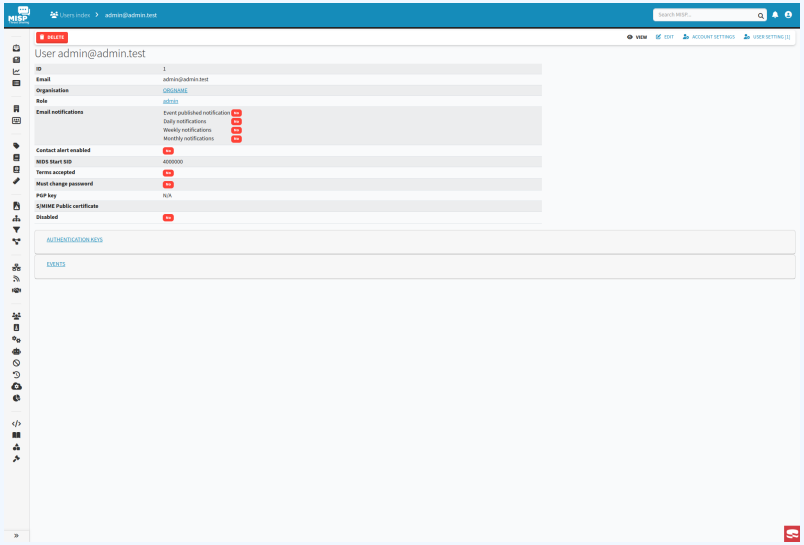


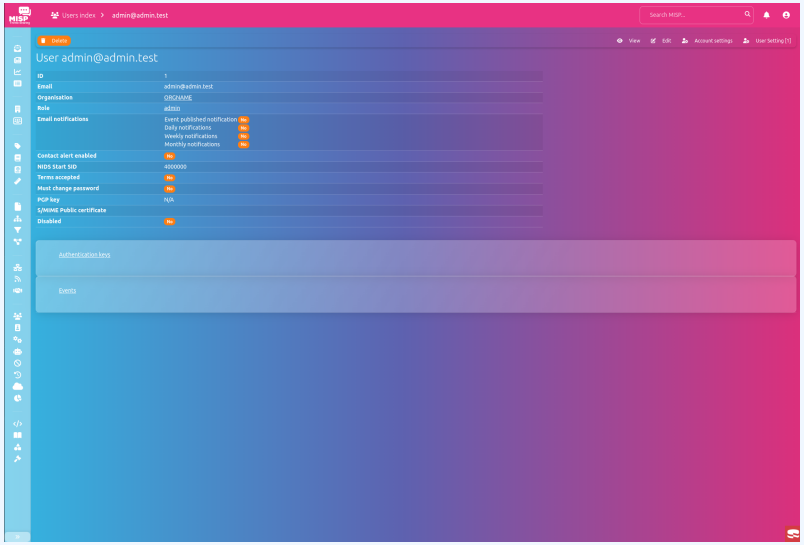
Codebase Migration: Look and Feel II

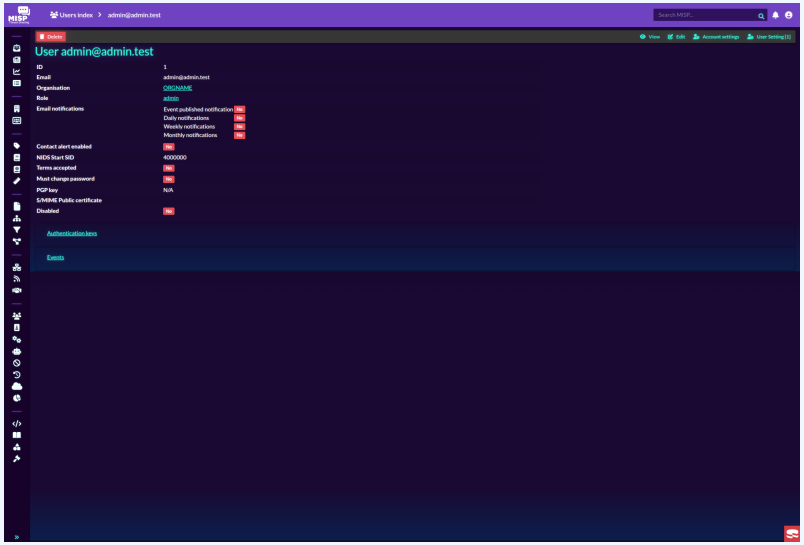
- Updating Bootstrap greatly improves aesthetics
- And allow us to integrate themes seamlessly

Codebase Migration: Look and Feel II





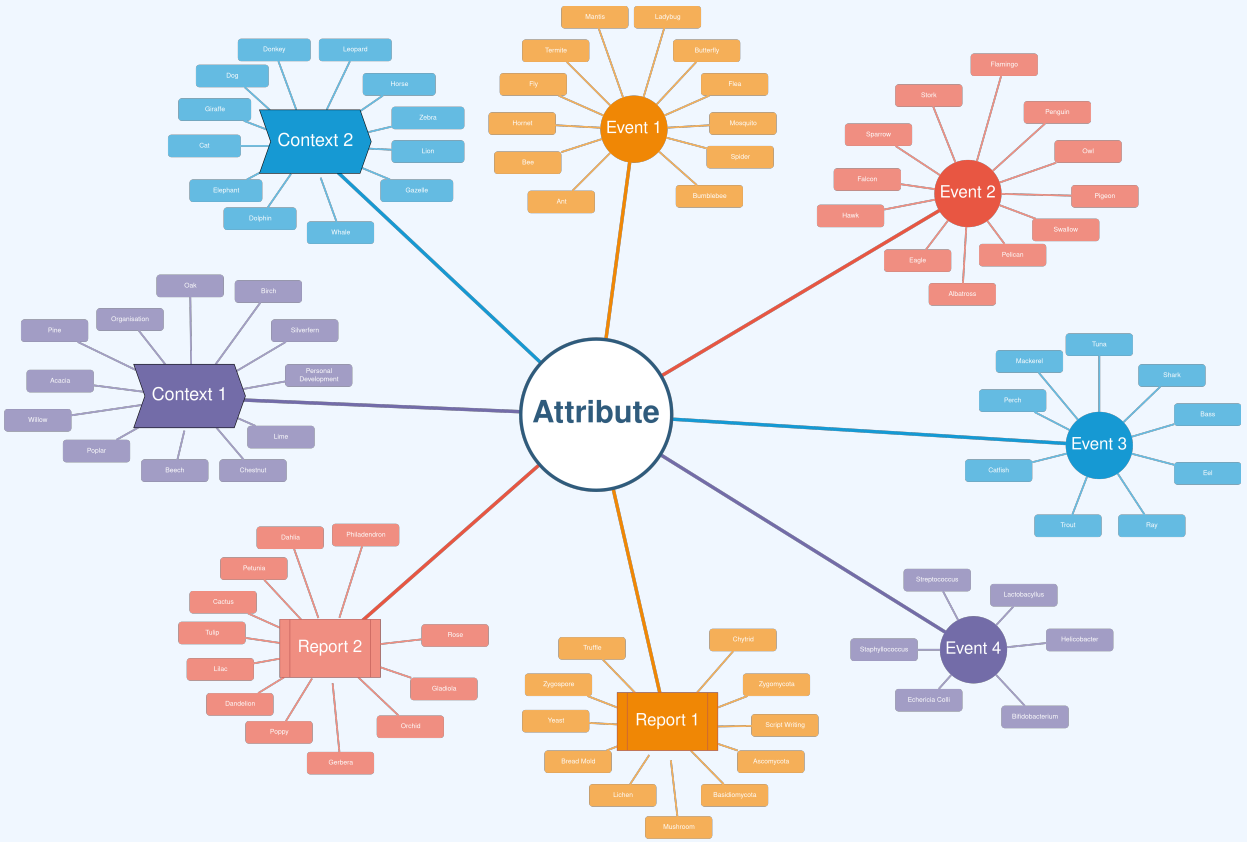




Step III - The TODOs

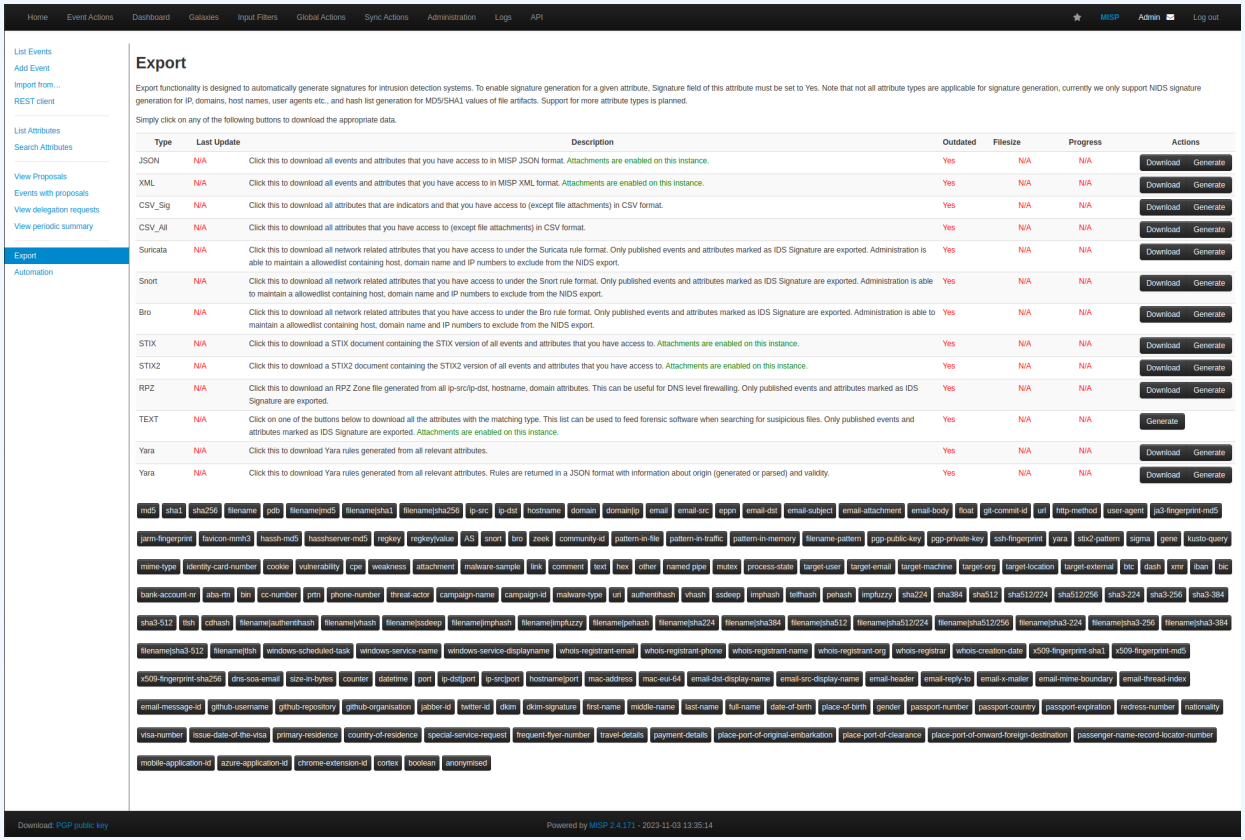
Redefine how we interact with data I

- Indicator centric perspective
  - Unified view of everything we know about a given Indicator
  - Allows us to take better decisions
  - Enable users to manage their IoC working set
  - Start an investigation more easily from a single indicator



Redefine how we interact with data II

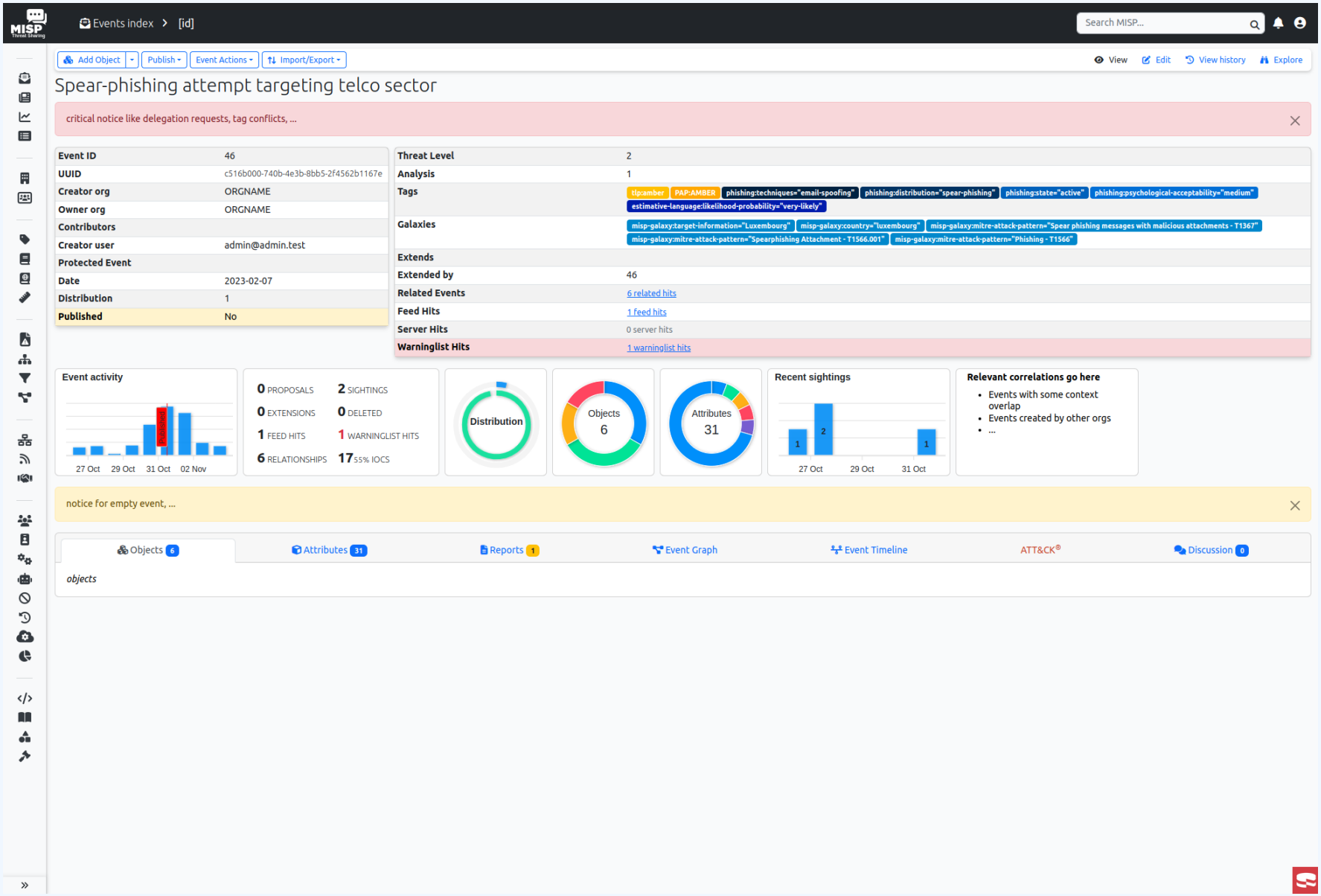
- Unified search mechanics
  - Code deduplication
  - Streamlined way to search for data
  - Opening up the full power of the API searches to UI users
  - Translation layer for the deprecated endpoints



Redefine how we interact with data III

- Refactor the Event view
  - Key Elements at first glance
  - Emphasis on the context (Insights, Taxonomies, Galaxies, Correlation, ·)
  - Massive performance gains by moving to the composition of separate atomic endpoints
  - Sneak peak ? ☻

Sneak peak of the new Event view - WiP



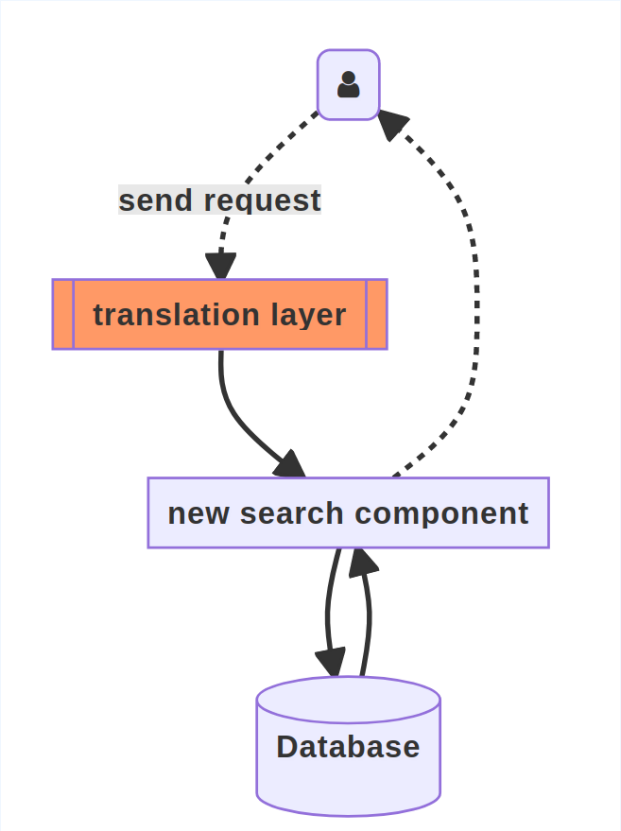
Considerations

MISP Core format

- Created in **2012**, Officially became a standard in 2016
- **No breaking changes** since its birth, And we'll maintain this streak
- Format will keep evolving to support new functionalities

API Compatibility

- The aim is to achieve a **near 100% compatibility** with the old API
- "Near" only due to the functionalities removed as a result of deprecation.
- Strategy: Mapping with a translation layer



Synchronisation compatibility

- API Compatibility means Synchronisation compatibility
- MISP 3 servers will be able to sync with MISP 2.4 and vice versa

BUT

- MISP **2.4** → **3**
  - Full support
- MISP **3** → **2.4**
  - Lossy when sharing new types of datapoints
  - E.g: Tags on Objects

Support for MISP 2.4

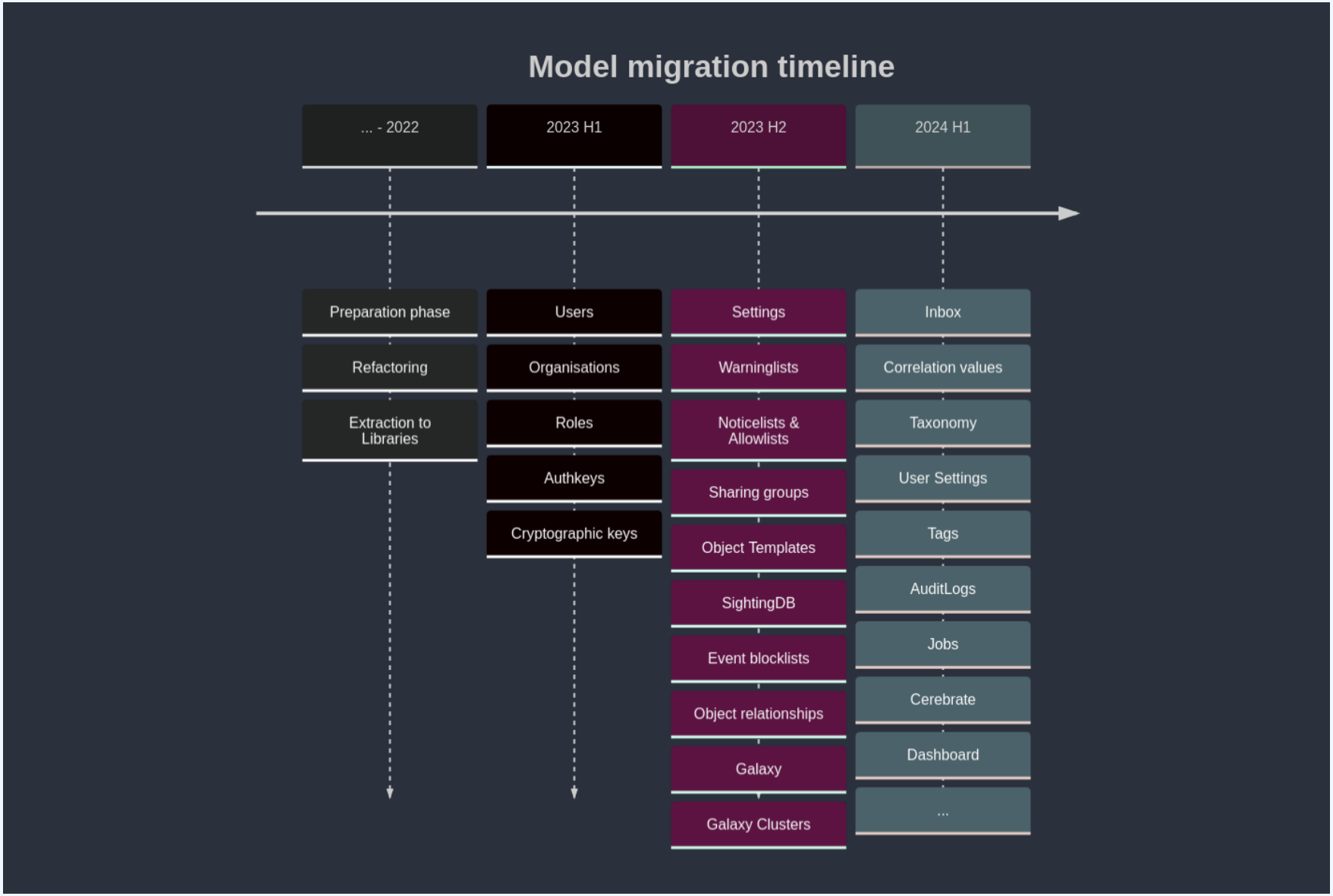
- MISP 2.4 will be **supported for a limited time**
- **6 months** support post MISP 3 release
  - Potential changes/improvements on 2.4 to better support MISP 3 interactions

Migration support for 2.4 → 3

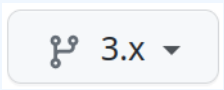
- MISP 2.4 will be **supported for a limited time**
- **6 months** support post MISP 3 release
  - No one-click update; manual script execution required
  - Migration tools will be included in MISP 3 to help you
  - This allows us to make underlaying changes such as
    - Database changes
    - Libraries changes (e.g supervisor in favour of cake-resque)

Installation for new instances

- **Simplified** installation based on package managers
- Upstream Docker installer
- OS targerts: **Ubuntu** and **RHEL**


Model migration timeline
... - 2022
2023 H1
2023 H2
2024 H1
Preparation phase
Refactoring
Extraction to Libraries
Users
Organisations
Roles
Authkeys
Cryptographic keys
Settings
Warninglists
Noticelists & Allowlists
Sharing groups
Object Templates
SightingDB
Event blocklists
Object relationships
Galaxy
Galaxy Clusters
Inbox
Correlation values
Taxonomy
User Settings
Tags
AuditLogs
Jobs
Cerebrate
Dashboard
...

Our hopes and expectations for the FIRST community

- We will list features marked for culling
  - If you're using any of them, please let us know!
- We will be lauching a beta phase in the future
  - Feedback & improvements are more than welcome!
- Want to get involved?
  - 

3-x branch - `MISP/MISP/tree/3.x`
  - Project for migration - `github.com/orgs/MISP/projects/2`