

# Customizable Dashboards



## Planned feature

This page describes a feature planned for future midPoint versions.

This feature is roughly designed and it was evaluated as feasible. However, there is currently no specific plan when it will be implemented because there is no funding for this development yet. In case that you are interested in [supporting](#) development of this feature, please consider activating [midPoint Platform subscription](#).

## Motivation

From the early days of midPoint there were two hardcoded dashboards: end-user home dashboard and administration dashboard. That was fine while midPoint was a simple IDM system. But now we are into identity governance, compliance, risk management support and other business-oriented aspects of identity. Whenever a "business" is involved there is a need for quick and clear presentation of data. Therefore the old hardcoded dashboards are no longer adequate for the job.

## More Dashboards

When it comes to data presentation, there is no single way how to do it right. Security officer will be interested certain specific aspects of data, but that aspects make little sense for a business manager. And then data protection officer will need yet another way to present the data. Therefore we need something that will be flexible and customizable. Such as customizable dashboards.

The idea is to make dashboard a first-class citizen in the midPoint universe. Each deployment can have as many dashboards as needed, and they can contain custom "widgets" used to present interesting pieces of information. Simple widgets can present data from [object collections](#), such as number of active employees, number of failed resources, suspended tasks and so. That is an elegant way to create *status dashboards* showing status of important system components. But collections are much more powerful, especially the ability to define collection *domain*. Therefore dynamic dashboard can be used for [compliance](#) purposes. Data protection dashboards may be set up. And so on.

Dashboard widgets could be customized. Both their look&feel and behavior. They can display custom data, adapt colors to show warnings and so on.

This is all a preliminary plan. But there is already some experimental functionality to validate those concepts. Please see [Dashboards, Widgets and Reports Design Notes](#) page to get a more in-depth technical information about our plans.

## Dashboards and Reports

Reports and dashboards are two almost independent things. Both work on the same data, but reports are static (printable) while dashboards are dynamic (interactive). But there is one use case when dashboards and reports work together: status reports.

System administrators often like to get regular mail summary of system operation during last 24hours. But we already have a dashboard functionality for the same thing. And here in the midPoint land we simply hate to duplicate functionality. Therefore what we need is a printable and mailable version of status dashboard. And that is exactly what status report is about. The report takes status dashboard configuration as an input. But instead of displaying dashboards in an interactive way it renders them in a stand-alone static way. Such as HTML table that can be delivered in regular mail message.

## Object Collections and Policies

Dashboards are meant to present data. But their integration with [object collections](#) brings interesting possibilities. In midPoint 4.0 several objects were "promoted" to be able to contain assignments. Object collection was one of those. This means, that object collections can now contain policy rules. Which means that there may be policy rules that can trigger on the number of objects with specific characteristics. E.g. a policy rules may trigger if there are more than 5% resources that are down. Result of this policy rules can be displayed in the dashboard. But there may be more interesting configurations with more substantial effects. Please see [Object Collections and Views](#) for more details.

## See Also

- [Object Collections and Views](#)
- [Compliance](#)
- [Archetypes](#)