



Project acronym: **midScale**

Project title: **MidPoint Scalability**

Third Parties: **Evolveum, s.r.o.**



Final Report

MidPoint Scalability

Authors:	Igor Farinič, Radovan Semančík, et al.
Submission date:	14 th October 2021
Dissemination level:	Confidential

Abstract: Automated identity management is a necessary condition for implementing identity governance and data protection policies in most large organizations. MidPoint is a leading open source identity management and governance platform. However, midPoint was originally designed for medium-size organizations. Limited scalability capabilities of midPoint posed an obstacle for large-scale midPoint deployments. MidScale project aimed at overcoming this obstacle. Re-implementing midPoint data store layers, significantly improving clustering capabilities, visibility and manageability, focusing on stability and quality assurance – all of that significantly improved scalability possibilities of midPoint. An open source identity governance platform can now be deployed at scales well beyond millions of managed identities. This capability enabled midPoint to address large customers, especially in academia and government where open source nature of identity management platform is crucial. Improved scalability also allows expansion into cloud service markets.



Disclaimer

The sole responsibility for the content of this publication lies with the authors. It does not necessarily reflect the opinion of the European Commission. The European Commission is not responsible for any use that may be made of the information contained therein.

Copyright

This document may not be copied, reproduced, or modified in whole or in part for any purpose without written permission from the NGI Consortium. In addition, an acknowledgement of the authors of the document and all applicable portions of the copyright notice must be clearly referenced. All rights reserved.

This document may change without notice.

Table of contents

1	Project aims and objectives.....	1
2	Work done in the project.....	1
2.1	Achievement of milestones.....	1
2.2	Deliverables.....	3
2.3	Results, exploitation and impact.....	4
2.4	Intellectual property rights.....	4
2.5	Promoting NGI_TRUST.....	4
3	Financial report.....	5
	Appendix.....	8



1 Project aims and objectives

The aim of midScale project is to increase scalability of midPoint at least by one order of magnitude. The goal is to gain capability to routinely manage millions of identities and making deployments with tens of millions of identities feasible. This improvement will make identity management and data protection features of midPoint available to wide range of organizations, which we believe will provide a significant income for further development and maintenance of midPoint.

MidPoint was designed with replaceable data storage components and this design decision allowed us to make midPoint scalable. We re-implemented midPoint data store mechanisms to support large data sets and take full advantage of PostgreSQL open-source database. We also significantly improved clustering capabilities and distributed task management, to allow features such as autoscaling, that make midPoint more at home in cloud environments. Important part of the plan was a focus on stability and quality assurance to make sure that midPoint is ready for large-scale deployments. Specialized testing environment was created for that purpose, and a user interface testing framework (“Schrodinger”) was significantly extended. The overall goal was to improve scalability of midPoint at least by one order of magnitude, reaching millions to tens of millions of managed identities.

The objectives of the midScale project include following improvements of midPoint:

- Objective 1: Improve scalability of midPoint at least by one order of magnitude, reaching millions to tens of millions of managed identities (overall scalability objective)
- Objective 2: Identify scalability and performance obstacles
 - Design and build performance testing environment.
 - Execute tests, analyse results.
- Objective 3: Overcome scaling obstacles
 - Scalability of data store (repository)
 - Scalability of distributed tasks (clustering)
 - Visibility, diagnostics and manageability

The core midScale team has 7 members: Katarina Bolemant (Valalikova), Pavol Mederly, Richard Richter, Anton Tkacik, Igor Farinic, Patrik Pilisi and Kateryna Honchar. First six members all Evolveum’s employees. Kateryna is affiliated with Evolveum, due to legal reasons we cannot employ her yet.

Making sure delivering the objectives, we have committed more of our resources that are not part of official midScale team, namely: Radovan Semancik, Kamil Jires and Jan Mederly.

There were also other 4 non-technical resources allocated to help with the load of different affiliated NGI Trust activities.

2 Work done in the project

MidScale project was based on multi-layered approach, addressing scalability challenges from many perspectives.

The key element of the solution was a new data store (repository) implementation, relying on open-source PostgreSQL database. Previous repository implementation was very generic, supporting several databases. This approach imposed significant limitations, as the implementation can effectively use only the characteristics that were “standard” and common to all supported database engines. Additional abstraction layer (Hibernate) introduces further limitations. Therefore, it was decided to replace the implementation with a new implementation (nick-named “sqale”), based on native approach. New implementation relies directly on PostgreSQL database, utilizing features specific to PostgreSQL to gain maximum performance and scalability advantage. New database schema (data model) was designed, a schema that takes advantage of PostgreSQL-specific mechanisms. Implementation of this particular component was perhaps the most challenging part of the project. Despite that, the resulting implementation exceeds expectations.

Although the repository part was clearly a key part of the solution, it could not provide expected scalability characteristics just by itself. Distributed task management system of midPoint was significantly improved, with some parts completely re-worked. New implementation improved abilities to distribute workload among many nodes. There were performance improvements in numerous parts of the system.

Large deployments are not just about low-level scalability and performance. The system does not need just “run”, it also has to be managed and monitored. Therefore, improvements to visibility, monitoring and manageability were an integral part of the project. Diagnostics capabilities were improved, to enhance visibility and manageability of large midPoint deployments. Parts of user interface were improved, providing better user experience, allowing easier management of large user bases. New query language (Axiom query language) was developed, enabling construction of complex, yet human-readable queries.

Last, but definitely not least, was the quality assurance part of the project. New performance testing environment was built from the ground up.

Based on the design workshops and collected requirements our preferred choice for the nature of the tests was to go for dynamic environment (Kubernetes) with lego-like approach to build and reuse testing components (pods).

We have configured the lab environment whenever applicable to follow modern GitOps principles. All configuration is stored in Git and kept synchronized by the Flux component to the Kubernetes cluster. All changes in configuration are properly reviewed in git before applied to the environment.

We are continuously running following type of tests:

- Component integration performance tests - these are like advanced unit tests, where we are running them in maven-controlled environment. They are written and maintained by developers to have quick feedback loop about performance improvements in the code. Some of the tests even when isolated are executed for up to 20 hours to collect enough data to



identify bottlenecks and further code optimization. These are executed from Jenkins using maven and are using our PSQL test results DB.

- UI performance tests - a simple suite to monitor our progress in UI typical page rendering times and number of DB requests. We have been using those tests to lower the DB load and speed up page rendering.
- UI feature tests a.k.a. Schrodinger tests - we have built a UI feature testing coverage on top of Selenide, automating our trainings, but mostly provide a reliable method to stabilise the UI.
- Automated end-to-end performance tests - Typical scenario built into end-to-end automated tests, usually running for long hours/days and working with huge amount of data in an integrated environment with typical systems connected to midPoint.
- Manual end-to-end performance tests - for most of the tests we are using templates from automated end-to-end perf tests, but we are performing also some manual steps and evaluation. Typical scenario is midPoint cluster auto-scaling. Were we are adding or removing nodes from existing running cluster under different loads and observing automated recovery from error states.

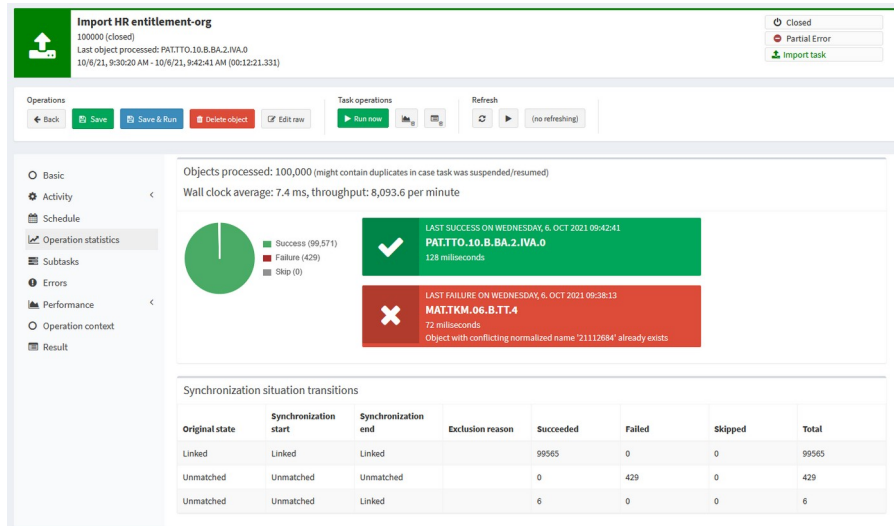
The testing environment was used in earlier phases of the project to evaluate midPoint and identify performance and scalability issues. The same environment was used later in the project to validate new components and improvements, and to assess the extent of performance and scalability improvements achieved by midScale project. The testing environment itself is a precious asset. We plan to maintain the environment in a long term, even extend it to provide better quality assurance for future versions of midPoint.

Major achievements of midScale project:

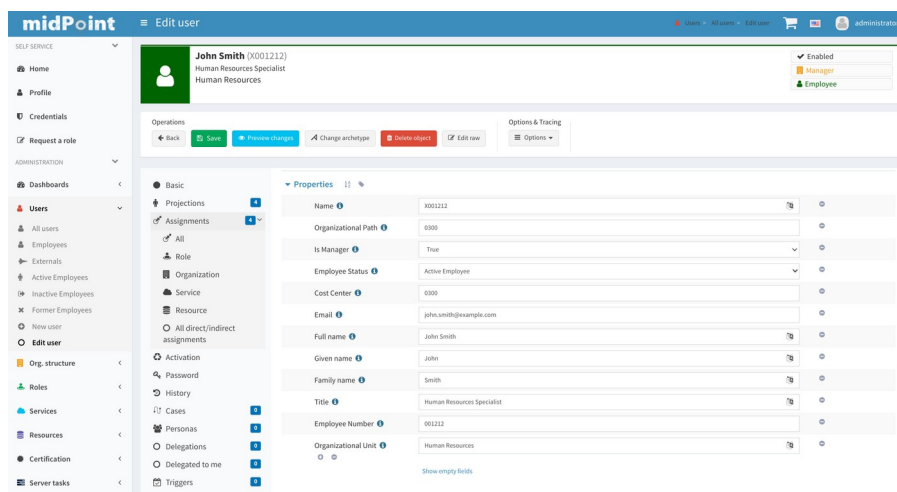
- Native PostgreSQL repository implementation (internal code name “sqale”). New implementation is based directly on PostgreSQL database, enabling use of PostgreSQL-specific features, avoiding inefficiencies introduced by excessive abstractions of previous database-agnostic implementation.
- Improved database schema (data model), focusing on efficient data storage and retrieval (a.k.a. “queries”). Queries that were not feasible in previous midPoint versions are currently feasible, even with a considerably larger data set.
- Axiom query language: human-readable query language, used in large/complex reports and advanced system administration, usually needed in large-scale deployments.
- Task management system was significantly reworked, especially focusing on distributed (clustered) tasks: introducing concept of flexible “activity”, significantly improving error detection and handling, major visibility improvements, faster management of distributed tasks. These improvements were necessary enabler for autoscaling and similar cluster-wide functionality. Improved visibility is essential for management of large, multi-node deployments.



- Autoscaling: ability to dynamically adapt midPoint performance (tasks) to changed number of nodes in midPoint cluster.
- Numerous improvements of visibility and diagnostics, significantly improving capability of issue diagnostics in large deployments.



- Improved user experience of graphical user interface (GUI), allowing more convenient administration of large and complex deployments.



- Quality assurance: dedicated performance testing environment is used to execute complex test scenarios, including GUI testing featuring Schrodinger testing framework.

Other achievements of midScale project:

- Analysis of midPoint, identifying performance issues and scalability obstacles.
- Partitioning of audit database table enables long-term storage of audit records in midPoint database, allowing to efficiently discard old data.
- Improvements to midPoint internal repository interface, allowing new modes of operation with large data (iterative/dynamic processing).
- Support of JSON in the repository (speed-up and storage efficiency)



- Optimized Prism data representation library, when working with data containers and large number of values.
- Eliminating numerous inefficiencies in midPoint code, mostly in Prism, Projector and Clockwork, components that are essential for computation and transformation of identity data in midPoint.
- Resolving notorious thread safety problems that riddled previous midPoint versions, limiting scale of midPoint deployments.
- Optimized graphical user interface (GUI) to execute less database queries, making it scalable for more users.
- Trainings of team members were conducted, filling the gaps in team knowledge about PostgreSQL database.
- Schrodinger testing framework was significantly improved, both the breadth and the depth of the testing was significantly improved. Numerous new tests were added, improving long-term maintainability of midPoint. The framework was made more flexible, the programming interface is more fluent and readable, contributing to readability and maintainability of the tests. The framework was separated to a dedicated project, motivating re-use (e.g. to test midPoint deployments).
- Dynamic auto-configurable testing environment based on Kubernetes and Ceph, following modern GitOps principles.
- Continuous execution of 5 different major tests suites. Prepared, set-up for execution and evaluation during the project.
- User survey was conducted during the project, to validate the project plan. Although there were relatively low number of responses, the responses were mostly positive, confirming our plans for MidScale project.
- Technology workshops were conducted, introducing midScale functionality to midPoint community. Workshop slides and recording were published using Evolveum communication channels.
- Blog posts were published and communicated using Evolveum marketing channels.

Work done on top of original plan:

- User experience improvements of midPoint GUI went beyond the original plan, resulting in usable, consistent and flexible user interface implementation.
- Prism data representation library was separated from main midPoint code, bringing it closer to being a generic re-usable component (continuation of Axiom-related effort started in midPrivacy project).
- Reports and dashboards functionality was improved, which brings additional benefits to visibility and manageability of large-scale deployments.
- Cluster-wide thresholds, enabling safe execution of cluster-wide tasks.
- Connector autoloading mechanism was introduced, allowing dynamic loading of new connectors, reducing need for system restarts and downtime.



Performance/scalability improvement results, charts

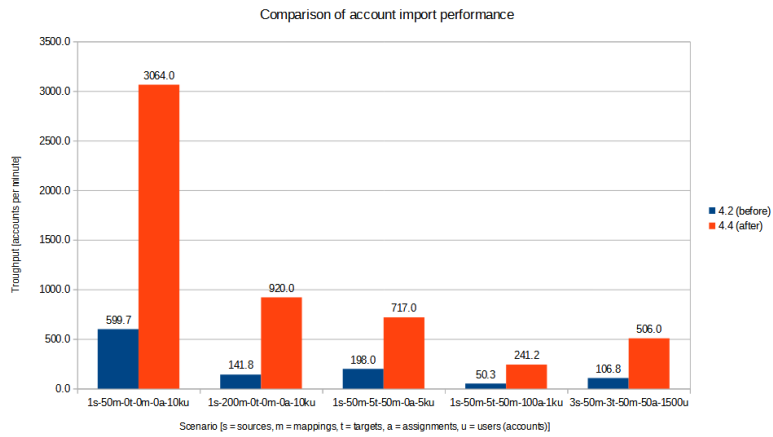


Figure 1: Internal code performance evaluation for account import for different input set scenarios. Increase in performance 262-549%

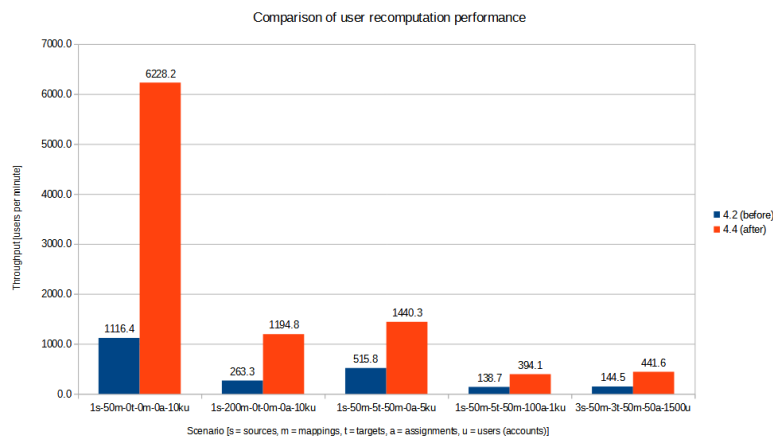


Figure 2: Internal code performance evaluation for user recomputation scenario for different input set scenarios. Increase in performance 205-458%

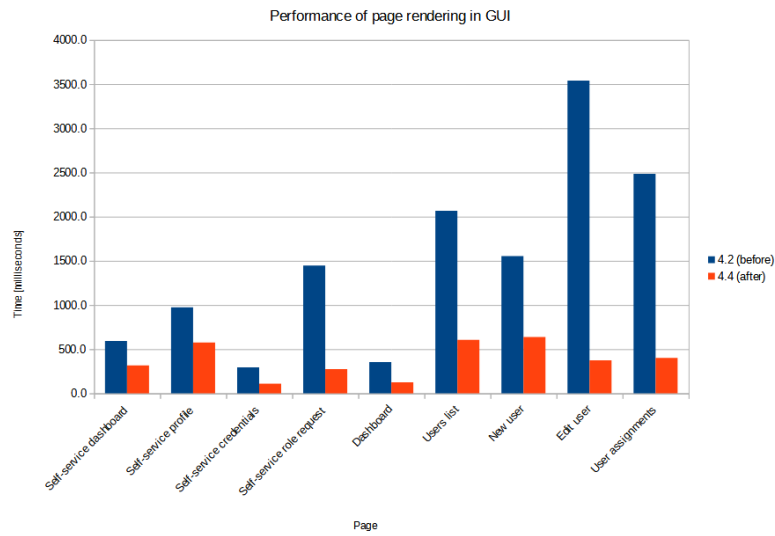


Figure 3: Performance gains for UI page rendering. Different pages. Performance increase 80-840%

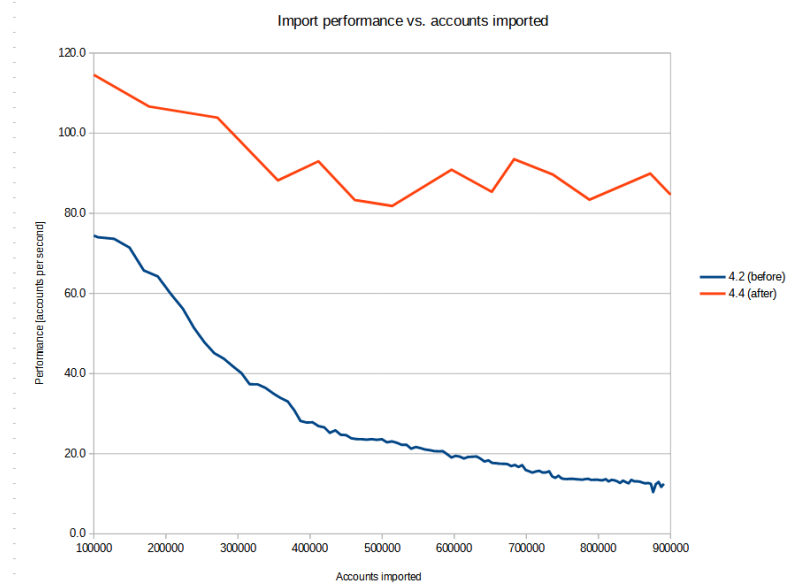


Figure 4: End-to-end testing in real environment. Same configuration, unlimited hardware resources, NVMe disks

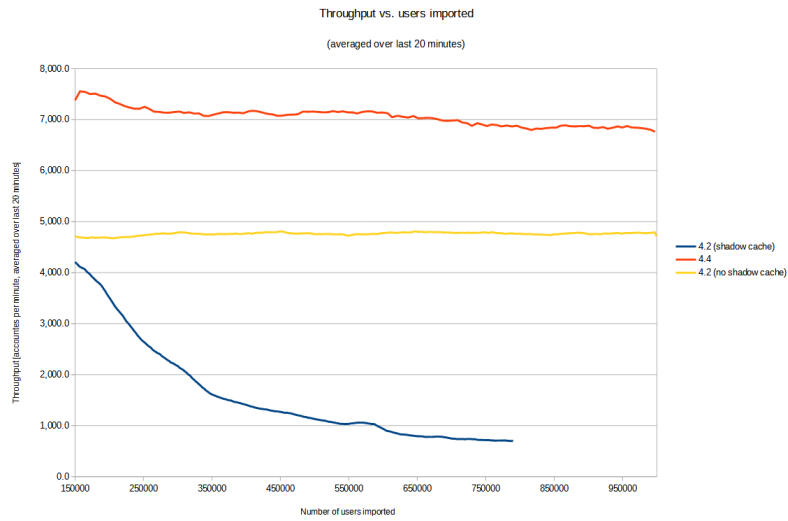


Figure 5: End-to-end testing in real environment. Leveling the field, disabled caching (non-scalable functionality) only for 4.2. Needed in typical env, but not for this scenario. Unlimited hardware resources, NVMe disks

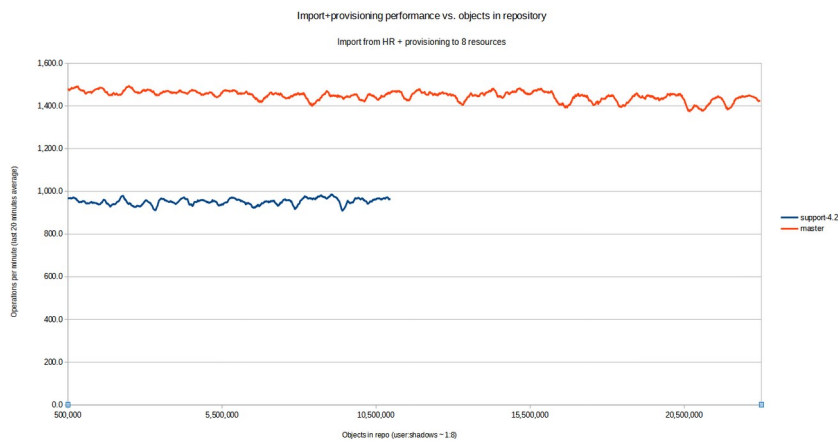


Figure 6: End-to-end testing in real environment. Leveling the field, disabled caching (non-scalable functionality) only for 4.2. Needed in typical env, but not for this scenario. Unlimited hardware resources, NVMe disks, code improvements in a complex setup

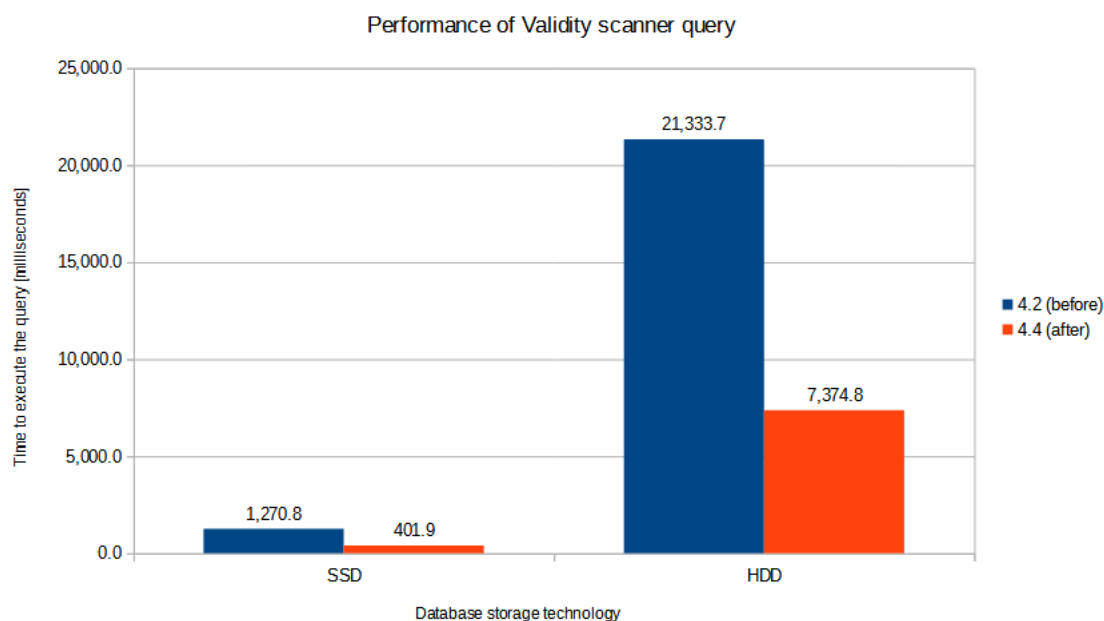


Figure 7: End-to-end testing focused on problematic scenarios “before”. Also focus on hardware resources “unlimited” vs constrained. Performance gain 189-216%. Little bit better gains when more and better hw resources are allocated.

2.1 Key project milestones

Table 1: Overview of milestones

Milestone number	Title	Date achieved
START	Project start	15 Oct 2020
MS1	Performance environment (milestone outcomes report)	26 Nov 2020
MS2	Performance evaluation and repository analysis/design (milestone outcomes report)	15 Jan 2021
MS3	Multithreading, Schrodinger and Query language (milestone outcomes report)	26 Feb 2021
MS4	Performance Repo prototype, UI basic tests, Multinode Tasks (milestone outcomes report)	12 Apr 2021
MS5	PostgreSQL, Performance environment (2), UX Analysis & Design (milestone outcomes report)	31 May 2021
MS6	Repository optimization, performance evaluation (milestone outcomes report)	9 Jul 2021
MS7	Migration Procedure, GUI Improvements, Auto-scaling (milestone outcomes report)	31 Aug 2021

Milestone number	Title	Date achieved
MS8 (FINISH)	Project finish (milestone outcomes report)	14 Oct 2021

Table 2: Milestone 1 - Performance environment

MS1	Performance environment	Start month : M1	End Month: M2
Objective	Design and prototyping of performance environment. Design and prototyping of selected parts of the solution were performed in this milestone, ahead of original plan.		
Related Deliverable	D1 - Architecture and Design documentation (also see milestone outcomes report)		
Participants	Igor Farinic, Patrik Pilisi, Pavol Mederly, Kate Honchar		
Person Months	7,8 PM		

Table 3: Milestone 2 - Perf evaluation and repository analysis/design

MS2	Perf evaluation and repository analysis/design	Start month : M2	End Month: M3
Objective	Performance evaluation of midPoint, identification of performance and scalability issues Design of repository (data store), using native PostgreSQL approach. Design of query language, Schrödinger testing framework and some other tasks were performed in this milestone, ahead of original plan.		
Related Deliverable	D1 - Architecture and Design documentation (also see milestone outcomes report)		
Participants	Anton Tkacik, Pavol Mederly, Richard Richter, Katarina Bolemant, Kate Honchar		
Person Months	8,7 PM		

Table 4: Milestone 3 - Multithreading, Schrodinger and Query language



MS3	Multithreading, Schrodinger and Query language	Start month : M3	End Month: M5
Objective	<p>Multithreading and robustness improvements.</p> <p>Schrödinger testing framework: work on the framework continued, improving the framework and adding new tests.</p> <p>Axiom query language was implemented and integrated into midPoint code.</p> <p>Work on performance testing environment continued, further improving the environment.</p> <p>Work on native PostgreSQL repository prototype continued after the design was finished.</p> <p>Conducting a user survey (started in this milestone).</p>		
Related Deliverable	<p>D1 - Architecture and Design documentation</p> <p>D2 - Axiom query language documentation (also see milestone outcomes report)</p>		
Participants	Anton Tkacik, Pavol Mederly, Kate Honchar, Richard Richter		
Person Months	14,8 PM		

Table 5: Milestone 4 - Performance Repo prototype, UI basic tests, Multinode Tasks

MS4	Performance Repo prototype, UI basic tests, Multinode Tasks	Start month : M5	End Month: M6
Objective	<p>Work on native PostgreSQL repository prototype continued, culminating in midPoint 4.3 release.</p> <p>User interface testing using Schrödinger testing framework which started in previous milestones culminated in this milestone, being integrated into the CI/CD mechanisms in performance testing environment.</p> <p>Multinode Tasks. There were significant improvements in user interface and visibility for multi-node tasks. Some work on multi-node tasks and task management in general was done in previous milestones, ahead of the plan.</p> <p>Work on performance testing environment continued, further improving the environment.</p> <p>User survey was finished, close, the results were analyzed and published.</p>		
Related Deliverable	<p>D3 - MidPoint 4.3 "Faraday" release</p> <p>D4 - Results of midScale survey. (also see milestone outcomes report)</p>		
Participants	Richard Richter, Anton Tkacik Pavol Medery, Katarina Bolemant, Kate Honchar		
Person Months	18,5 PM		

Table 6: Milestone 5 - PostgreSQL, Performance environment (2), UX Analysis & Design

MS5	PostgreSQL, Performance environment (2), UX Analysis & Design	Start month : M6	End Month: M8
Objective	<p>Work on native PostgreSQL repository implementation continued, based on the prototype code released in midPoint 4.3.</p> <p>Performance testing environment was mostly completed in MS5, first comprehensive performance tests were prepared and executed in the environment.</p> <p>Task management improvements were proposed during MS5, ahead of plan, as a preparatory work for implementing task autoscaling capabilities.</p> <p>User experience (UX) analysis and design was conducted, several improvements were proposed.</p> <p>Technology workshops regarding new midPoint 4.3 functionality were conducted.</p> <p>User interface testing using Schrödinger testing framework continued, further extending the framework and adding tests.</p>		
Related Deliverable	<p>D5 - Technology workshops, slides and recordings.</p> <p>D6 - Overview of performance testing environment (also see milestone outcomes report)</p>		
Participants	Pavol Mederly, Richard Richter, Katarina Bolemant, Kate Honchar, Igor Farinic, Patrik Pilisi, Anton Tkacik		
Person Months	2,3 PM		

Table 7: Milestone 6 - Repository optimization, performance evaluation

MS6	Repository optimization, performance evaluation	Start month : M8	End Month: M9
Objective	<p>Work on native PostgreSQL repository implementation continued.</p> <p>Performance evaluation was performed, focusing mostly on code of projector component.</p> <p>Task management improvements. Task management improvements proposed during MS5 were implemented in MS6.</p> <p>Connector autoloading was implemented, on top of original plan.</p> <p>User interface testing using Schrödinger testing framework continued, further extending the framework and adding tests.</p>		
Related Deliverable	(also see milestone outcomes report)		
Participants	Richard Richter, Pavol Mederly, Anton Tkacik		



MS6	Repository optimization, performance evaluation	Start month : M8	End Month: M9
Person Months	12,3 PM		

Table 8: Milestone 7 - Migration Procedure, GUI Improvements, Auto-scaling

MS7	Migration Procedure, GUI Improvements, Auto-scaling	Start month : M9	End Month: M11
Objective	<p>Migration procedure was addressed, implementing the designed capabilities, testing migration from midPoint 4.0 to midPoint 4.4, testing migration procedure for changing database implementations.</p> <p>Auto-scaling mechanisms were developed in MS7 (based on improvement done in previous milestones, ahead of plan), allowing midPoint cluster to scale automatically in cloud environment.</p> <p>User experience improvements were implemented in various parts of midPoint user interface.</p> <p>Work on native PostgreSQL repository implementation was finished in MS7, with some minor leftovers.</p>		
Related Deliverable	(also see milestone outcomes report)		
Participants	Richard Richter, Kate Honchar, Pavol Mederly		
Person Months	8.1 PM		

Table 9: Milestone 8 - Project Finish

MS8	Project Finish	Start month : M11	End Month: M12
Objective	Release Acceptance testing, Solution review, Documentation		
Related Deliverable	D7 - MidPoint 4.4 "Tesla" candidate release (also see milestone outcomes report)		
Participants	Pavol Mederly, Richard Richter, Katarina Bolemant, Anton Tkacik, Kate Honchar, Igor Farinic, Patrik Pilisi		
Person Months	0,9 PM		

There were numerous challenges in the project. Some of them were related to technology, however, there were also organizational challenges:

- The effort required to implement new repository was much higher than expected. This was mitigated by adding more resources to the project.

- Testing environment was much more complex than expected, due to a use of more powerful technologies than originally planned. This challenge was addressed by adding a dedicated engineer to the project, completely funded by Evolveum.
- Specification and implementation of automated test cases was more difficult than expected. The challenge was addressed by basing some tests on existing test suite and test scenarios, extending the tests cover new cases and environments.
- Interpretation and analysis of test results were slightly challenging as well, requiring several iterations over test set-up and tuning to get relevant results.
- Many people worked on the project, naturally divided into several sub-teams. Coordination of work between the teams proved to be more challenging than expected.
- Perhaps the greatest challenge was to lead the project within the constraint given by the European community funding. Especially the rigidity of project plan, very limited possibility to re-plan and adjust the resources posed a major challenge for project leadership. There are always unforeseen circumstances, especially in innovative projects. The rigid constraints make it very difficult to dynamically address unexpected situations. At the end the project was executed more-or-less according to original plan. However, this caused unnecessary pressure on all team members, and it some of the opportunities that opened up during project execution might be lost due to lack of flexible options to pursue them.

2.2 Deliverables

Table 10: Overview of deliverables submitted/outputs produced

Deliverable number	Title	Date
D1	Architecture and design documentation (see MidScale Solution Architecture and MidScale Design Meeting Overview)	26 Nov 2020 (subsequently updated)
D2	Axiom query language documentation (published at Evolveum documentation site)	8 Apr 2021 (subsequently updated)
D3	MidPoint 4.3 "Faraday" release	9 Apr 2021
D4	Results of midScale survey (published on project site)	25 Mar 2021
D5	Technology workshops, slides and recordings (available at Evolveum "talks" page).	Apr-Jun 2021
D6	Overview of performance testing environment	19 May 2021



Deliverable number	Title	Date
		(subsequently updated)
D7	MidPoint 4.4 "Tesla" release candidate	14 Oct 2021

Every milestone in the project was followed by a corresponding milestone in midPoint development cycle. There is a source code snapshot (a.k.a. “git tag”) for each of the milestones. The results of midPoint build (a.k.a. “artifacts”) are published in Evolveum Maven repository (<https://nexus.evolveum.com/>).

2.3 Results, impact, mentoring and future exploitation

MidPoint deployments before midScale project were able to routinely handle tens or hundreds of thousands of identities. We had some success with deployments involving millions of identities, but such deployments usually require special treatment. Goal of midScale project was an increase in midPoint scalability at least by one order of magnitude. We have all the reasons to believe that midPoint deployment after midScale project could routinely handle environments with millions of identities, and that deployments with tens or even hundreds of millions of identities are possible.

The key results of the midScale project are:

- Key result 1: **Improved scalability of midPoint** at least by one order of magnitude, enabling deployment beyond millions of managed identities.
- Key result 2: Improved visibility, diagnostics and reliability of midPoint, making **long-term maintenance of large-scale deployments** feasible.
- Key result 3: Improved performance and user experience of midPoint user interface, making **management of massive user bases** efficient.

The midScale project was just the beginning for us.

We have built internal knowledge for building and operating infrastructure as a code methodology. Even during the midScale project, we have reached out to our partners, customers and community and started evaluating and building foundation for possibilities to built and operate IDaaS for our customers in cooperation with our partners. We had already opportunities on (un)conferences to run sessions to attract pilot customers to start pivoting. That is one of the business opportunities we haven’t predicted before midScale but matured thanks to the project. It opens us a whole new world of possibilities for growth and attracting even more customers.

For the original plan we have a long queue of waiting customers, we are in touch who are waiting to migrate to our new release based on the midScale results. Even for existing customers there are many benefits for simplified operation, saving cost (for hardware, ...). There are also many benefits for us, thanks to better visibility we would be able to provide faster and more effective support, thus keeping us more time to do even more for the community to attract more customers.



After the first wave of migrations, we would like to build more case studies with existing customers, to have more successful stories in large deployments and simplified migration and attracting more lucrative very large customers, we would be targeting more governmental institutions and municipalities.

Our marketing is very happy about the vast content we have generated for many campaigns for the upcoming months. We have already delivered a lot of webinars, blog and conferences in the first half of the project, but much more is to come in the coming months, webinars and video content proved to be very useful and we are going to continue and strengthen our position to have very fresh and relevant content also thanks to the project. There will be also other channels like classical blogging, conferences and so on.

We will keep the lights on also for continuing leveraging what we have built during the project, our automated tests suites and our testing environment to keep improving about scalability also in the future and also to build more test suites for more use cases (mostly integrations, end-to-end testing) to come.

We have originally anticipated to open the path to very large digital identity security market segment which we have confirmed by the project. However, it wasn't anticipated to open also market segment for Digital identity as a Service approach. Which we have already committed resources to go down this path even after the official midScale project ends. Therefore the market scalability, value and impact has much broader scope and implications than originally anticipated.

For the offered mentoring we were quite sceptical at first. We had a strong feeling it was strongly focused on getting investors on board. We as bootstrapped, open source committed company, with not so good experience finding potential investors in the past and lacking NGI initiatives communication the investors they are cooperating with have open source understanding were very careful to committed more resource to explore the possibilities.

We gave it a chance, explored the offering, were confused there were NGI Trust mentoring and NGI TETRA mentoring that were mostly overlapping. Reached out to both, decided to give a TETRA Bootcamp a try. We have found out there is more just to preparing for a pitch to investors and protecting IPR. Therefore we have committed even four more resources, non-technological people, from marketing and sales to Bootcamp and TETRA mentoring sessions after. It took several weeks, intensive work, mindmaps, exercises and we are happy we had undergo the experience. We have improved internally in sharing our vision, we are better in communicating it to the outside world, our people are more self-confident and more independent. We have prepared a lot of new marketing and sales materials, compiled and improved a lot of existing information, reached out gathered feedback from partners customers to many open questions, We have tried to reach out to more interesting leads and potential partners, we are trying now to fry much bigger fishes. There are still some open questions, especially how to keep the dialog open with very big solution providers, that came initially to us, are interested but are too big to keep the conversation flow and to build trust between organizations and not only people.

To summarize next steps, because we have so many as described above:

- Pivot opportunity for Identity as a Service with midPoint
- Attract very large customers from governmental and municipalities market segment.



- Continue execution of the plans with sales - going for more attractive partners, build better partnerships and work better with existing/new leads thanks to materials we have compiled.
- Marketing - support for sales team, which are now much better organized and have better understanding. Also have many materials to publish in coming months, going for more conferences, publishing more video content, ...
- Keep leveraging the deliverables like testing infrastructure to automate evaluation of scalability in the future and built more automated use cases.

2.4 Intellectual property rights

All deliverables of the midScale projects are publicly available.

All the code is available under the terms of Apache License 2.0 and/or European Union Public License (EUPL). This includes complete source code of midPoint, including all the improvements developed within the scope of midScale project.

The documentation is published at project home page (<https://docs.evolveum.com/midpoint/projects/midscale/>), available under the terms of Creative Common CC BY-NC-ND license.

There are some small parts of the project that are not published, mostly due to security reasons. This includes configuration details of the test environment, passwords, keys and similar secret information. Also, documents containing inappropriate amount of personal data regarding project team members are not published, or published in a limited form.

2.5 Promoting NGI_TRUST

We have promoted midScale project (and hence NGI_TRUST) using the usual Evolveum communication channels:

- Evolveum blog posts
 - <https://evolveum.com/introducing-midscale/>
 - <https://evolveum.com/midpoint-scalability-survey/>
 - <https://evolveum.com/midscale-survey-results/>
 - <https://evolveum.com/future-of-midpoint-4-3-and-4-4/>
 - <https://evolveum.com/midpoint-4-3-faraday/>
- Evolveum/midPoint mailing list communication
 - <https://lists.evolveum.com/pipermail/midpoint/>
- Public on-line workshops/webinars
 - <https://docs.evolveum.com/talks/>
- Evolveum YouTube channel
 - https://www.youtube.com/channel/UCSDs8qBlv7MgRKRLu1rU_FQ
- Internet2/InCommon communication channels (selective)
- NGI_TRUST 9th Results Webinar



2.6 Feedback and suggestion for NGI_TRUST and NGI overall

We are grateful for the opportunities offered by NGI_TRUST and NGI. We believe that such activities contribute to the goals of NGI initiative, and the society at large.

MidScale project certainly helped to improve the quality of our open-source product, opening up new interesting opportunities for the identity management community in general, and Evolveum business in particular.

While there are unquestionable benefits of NGI_TRUST programme, there are also some downsides.

Perhaps the most painful part of the project was the rigidity, imposed by the funding constraints. It was very difficult to keep the project running according to original plan, especially in the fact of unforeseen circumstances and situations. It was very difficult to adjust project plan, add more resources to the project, or even replace team members with more senior and more experienced engineers that were hired after the proposal was submitted. We have resolved the situation by “spending our own money” on top of original plan, adding people to the project that were completely funded by Evolveum. However, not all the organizations might be able to afford such a solution.

On one hand, NGI asks for *innovative* projects. On the other there are very limited means how to handle unforeseen situations, situation that are inherent to innovative projects. Our most emphasized suggestion would be to significantly increase the flexibility of the projects, allowing the project to adapt to changed circumstances.

We are technological company, sometimes forgetting also the world is also about other aspects, communication, marketing, sales and so on. So we had allocated all the budget to technical people. We understand most of it, but it would be helpful to get us better prepared what to expect. As we have stable business revenue stream we were able to adapt by allocating four more non-technical people, but for a typical start-up I could imagine it would be a lot of burden and not much they can do about it.

We would welcome the initial communication for the project to be summarized. There was some initial email, but not much was clear what to expect. There were more overlapping initiatives (mentoring from NGI TRUST vs NGI TETRA), most of them were not mentioned at all (interviews, questionnaires, feedbacks, conferences, talks, ...). There were some regular surveys or other activities, we were contacted by more organizations, we were not ready to expect such side loads. We were able to cope with most of the activities, by assigning more people, non-technical, aside of the original team.

Maybe even improve application guidance to expect and allocate part of the budget to non-technical activities (there is mention about it, but maybe emphasize it more by introducing them what to expect during the project), maybe have some estimates on your side and have it given. Those activities are essential, but one has to be prepared to make it through.

Innovation can come in many forms, we have sometimes felt the only feasible proposals are those mentioning very popular keywords. We are very happy it was not the case for NGI TRUST as for others initiatives and we were given this chance.



We would also welcome feedback for application where to improve, where we have lost points and so on. To have it even better in the future.

We would welcome to have clear communication about open-source compatible investors cooperating with NGI Trust.

We as bootstrapped company would also welcome alternative stream for this kind of companies. We see it difficult in today's world, the news is full of VC funding stories, putting successfully bootstrapped companies in the shadows. We are pretty sure there are more companies like ours, we do not have platform to meet and speak to each other. When combined with open source there are even fewer like this, but still there are and there was no space given to this type of organizations at all.

In our opinion, NGI should pursue following priorities:

- Personal data protection and GDPR compliance. GDPR is in force. However, it is very difficult for any organization to be completely GDPR-compliant. GDPR requires control and accountability level, which is almost impossible to achieve without automated tooling. However, such tooling is not readily available. Complex identity governance platforms are prohibitively expensive, while still not providing mechanisms for complete GDPR compliance. Even if an organization wants to be GDPR-compliant, full compliance may often be economically infeasible.
- Trustworthy and affordable identity governance. Identity management and governance are not just information security best practice, they form the backbone of information security. However, many organizations still have only a very rudimentary identity governance mechanisms in place. There are many initiatives that focus on authentication, federation, decentralized identity and similar cross-domain mechanisms, there is very little attention paid to the governance of identities *within* a domain. The organizations lack proper tooling, but more importantly, they lack appropriate expertise and practices. This leads to numerous security risks, data leaks, compliance issues and other problems.

3 Financial report

Information about the budgeted costs (per partner) and real costs incurred.

Table 11: Person-months

Name	Organisation	PM (plan)	PM (used)	% used
Katarina Bolemant (Valalikova)	Evolveum	2	6	300%
Pavol Mederly	Evolveum	6	9	150%
Richard Richter	Evolveum	10	11	110%
Anton Tkacik	Evolveum	11	12	109%
Igor Farinic	Evolveum	5	1.25	25%
Patrik Pilisi	Evolveum	4	6	150%
Kateryna Honchar	Evolveum	11	12	109%
TOTAL		49	57.25	116%

All deviations from the plan were properly communicated with NGI Trust representatives and approved by emails. Summary from the two email requests:

1. Would like to inform you about last name change of one of our team members: Katarina Valalikova is now Katarina Bolemant, as of her wedding.

We would like to also request the transfer of allocation between existing midScale team members:

Lower Igor Farinic allocation by 1.5 FTE and increase Katarina Bolemant allocation by 2.0 FTE. The request is due the type of midScale project work to be done where Katarina is more effective.

2. We would like to request the transfer of allocation between existing midScale team members:

Lower Igor Farinic allocation by 2.25 FTE and increase Katarina Bolemant allocation by 2.0 FTE, Richard Richter by 0.5 FTE and Patrik Pilisi by 2.0 FTE.

The request is due more technical knowledge was needed to finish the milestones on time, mostly optimise the internal code to scale better, finish new scalable repository implementation and to improve our testing environment.



Table 12: Budgeted and incurred costs

Item	Evolveum (plan)	Evolveum (used)	Total budget	Total used
Personnel costs	243 200,- Eur	258 587,89 Eur	243 200,- Eur	258 587,89 Eur
Equipment costs	6 000,- Eur	5 793,85 Eur	6 000,- Eur	5 793,85 Eur
<i>PostgreSQL training</i>	<i>6 000,- Eur</i>	<i>5 793,85 Eur</i>	<i>6 000,- Eur</i>	
Software licenses	0,- Eur	0,- Eur	0,- Eur	0,- Eur
Travel expenses	0,- Eur	0,- Eur	0,- Eur	0,- Eur
IPR	0,- Eur	0,- Eur	0,- Eur	0,- Eur
Total budget	249 200,- Eur	264 381,74 Eur	249 200,- Eur	264 381,74 Eur
Requested contribution	124 600,- Eur	124 600,- Eur	124 600,- Eur	124 600,- Eur



We have properly asked and got approved for re-allocation of the remaining budget from Equipment costs to Personnel costs:

I would like to request possibility to re-allocate portion of budget from Equipment cost - PostgreSQL training where we had allocation 6000,-Eur and actually spent 5793,85 Eur and would reallocate 206,15 Eur to Personnel cost.

Our Equipment cost - PostgreSQL training was an essential part to train our development team, because significant effort and future plans are to heavily rely on our Native midPoint implementation that is based on PostgreSQL. The training was delivered in online form by vendor of PostgreSQL DB itself the EnterpriseDB organization.



Horizon 2020 Programme
DG CNECT
Next-generation Internet



Appendix

As required