PROCEEDINGS OF SCIENCE

naco - the "n" attribute conformity checker

Dr. Marcus Hardt [a,*]

[a] *Karlsruhe Institute of Technology,*
Herrmann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany

E-mail: hardt@kit.edu

Federated identity management enables researchers to access services across organisational boundaries using their home institution credentials. However, the success of this model depends on consistent attribute release by Community Authentication and Authorisation Infrastructures (AAIs). When different Community AAIs release attributes inconsistently, service providers face increased development complexity, which ultimately threatens the adoption of federated identity in research. This paper presents NACO (N Attribute COnformity checker), a tool for automated, continuous validation of attribute compliance against specifications such as AARC-G056. NACO monitors multiple Community AAIs, validates attribute presence, type correctness, and scoping, and provides both command-line and REST API interfaces. We describe the design, implementation, and deployment of NACO across ten European and international Community AAIs, and discuss findings that have contributed to improved interoperability in federated research infrastructure.

International Symposium on Grids and Clouds (ISGC2026)
15 - 20 March, 2026
Academia Sinica Computing Centre (ASGC), Institute of Physics, Academia Sinica Taipei, Taiwan

*Speaker

© Copyright owned by the author(s) under the terms of the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC-ND 4.0) All rights for text and data mining, AI training, and similar technologies for commercial purposes, are reserved. ISSN 1824-8039 . Published by SISSA Medialab.

PoS(ISGC2026)032

1. Introduction

Modern research increasingly relies on international collaboration, requiring researchers to access computing resources, data repositories, and scientific services across institutional and national boundaries. Federated identity management addresses this challenge by allowing researchers to authenticate using credentials from their home institutions while accessing resources operated by different organisations.

At the heart of this model are Community Authentication and Authorization Infrastructures (Community AAIs), which act as intermediary proxies between Identity Providers (IdPs) at home institutions and the services that researchers need to access. These Community AAIs aggregate user attributes from home IdPs and release them to services in a standardised format. Services then use these attributes for access control decisions, personalisation, and audit logging.

1.1 The Need for Interoperability

The effectiveness of federated identity management depends critically on *interoperability*: the ability of services to work seamlessly with multiple Community AAIs, and conversely, the ability of users from any Community AAI to access any compatible service. This interoperability requires that Community AAIs release a consistent, well-defined set of user attributes.

In practice, however, attribute release varies significantly across Community AAIs. Different infrastructures may use different attribute names for the same semantic concept, release attributes in different formats (strings vs. lists), or omit certain attributes entirely. Home IdPs further compound this problem by providing attributes of varying quality and completeness to the Community AAIs.

These inconsistencies create substantial problems for service providers. Services must handle multiple attribute formats and fallback scenarios, increasing code complexity and maintenance burden. Some services may only work with certain Community AAIs, fragmenting the user experience. The difficulty of supporting multiple attribute profiles may even discourage service providers from joining federated infrastructures, leading to a poorer service landscape.

Ultimately, inconsistent attribute release threatens the success of independent identity management in research, potentially pushing communities toward centralised, proprietary solutions.

1.2 Contribution

This paper presents NACO (N Attribute COnformity checker), a tool designed to address the attribute compliance challenge through automated, continuous monitoring. NACO validates the attributes released by Community AAIs against configurable specifications such as AARC-G056 [1].

The key contributions of this work are: a flexible, JSON-based attribute specification system that supports mandatory and optional attribute sets, compound attributes, type validation, and scope checking; a dual-interface architecture providing both command-line and REST API access for integration with monitoring infrastructure; deployment experience from monitoring ten Community AAIs across European and international research infrastructures; and insights into common attribute compliance issues and their impact on interoperability.

The remainder of this paper is organised as follows: Section 2 provides background on federated identity and attribute specifications. Section 3 describes the design of NACO. Section 4 covers

PoS(ISGC2026)032

implementation details. Section 5 presents deployment experience and findings. Section 6 discusses future work, and Section 7 concludes.

2. Background

2.1 AARC Blueprint Architecture

The Authentication and Authorization for Research and Collaboration (AARC) project developed the Blueprint Architecture (BPA) [2] as a reference model for federated identity in research. The BPA defines a layered architecture where Community AAIs serve as proxy services between home IdPs and relying party services.

In this architecture, users authenticate at their home IdP (typically their university or research institution), which releases identity attributes to the Community AAI. The Community AAI may enrich these attributes with community-specific information (such as group memberships or entitlements) before releasing them to services. This proxy model allows communities to maintain control over authorisation while leveraging existing institutional identities.

The BPA supports multiple protocols, with OpenID Connect (OIDC) [3] being the predominant choice for new deployments. OIDC tokens carry user attributes as claims, which can be obtained from the UserInfo endpoint, embedded in the Access Token (for JWT-formatted tokens), or retrieved via Token Introspection.

2.2 Attribute Specifications

Several specifications define which attributes Community AAIs should release and in what format. AARC-G056 [1] (currently in draft status) specifies attribute profiles for research collaborations. It defines a basic attribute set that all Community AAIs should provide, including persistent identifiers, name information, email addresses, home organisation affiliation, and assurance levels. An extended attribute set covers additional attributes such as entitlements, ORCID identifiers [7], and SSH public keys. Infrastructures may define their own profiles based on AARC-G056, adapted to local requirements.

These specifications build on established schemas. The eduPerson schema defines attributes for the higher education community, including `eduPersonUniqueId`, `eduPersonAssurance` [4], and `eduPersonEntitlement` [6]. The VOPerson schema extends eduPerson for virtual organisations, adding attributes such as `voperson_id`, `voperson_external_affiliation` [5], and `voperson_verified_email`. The OpenID Connect Core specification defines standard claims such as `sub`, `name`, and `email`.

2.3 The Attribute Compliance Challenge

Despite these specifications, ensuring consistent attribute release across Community AAIs remains challenging. Community AAIs may implement specifications differently, choosing different attribute names from the allowed alternatives or using different data types (e.g., a single string vs. a list of strings). Furthermore, Community AAIs can only release attributes they receive from home IdPs, and many home IdPs do not release all requested attributes, or release them with incomplete

PoS(ISGC2026)032

or incorrect values. Even correctly configured systems may drift over time due to software updates, configuration changes, or changes in upstream IdP behaviour.

Prior to NACO, no automated tooling existed for continuous compliance monitoring. Compliance checking was typically manual, sporadic, and reactive rather than proactive. This gap motivated the development of NACO: a tool that can continuously monitor multiple Community AAIs and report on their attribute compliance status.

3. NACO Design

3.1 Architecture Overview

NACO is designed as a Python-based validation tool with two complementary interfaces. The command-line interface (`naco`) enables batch validation of configured Community AAIs, suitable for cron-based periodic monitoring and interactive debugging. It supports validating all configured AAIs or targeting specific ones, managing the validation cache, and generating reports. The web service (`naco-web`) provides a REST API built on FastAPI [8] that accepts validation requests for OIDC tokens and SAML assertions, enabling integration with external monitoring systems.

The core design principles are configurability (attribute specifications are defined in JSON files), multi-source validation (attributes are checked across multiple OIDC endpoints), continuous monitoring (results are cached and timestamped for trend analysis), and actionable output (reports clearly indicate which attributes are present, missing, or malformed).

3.2 Attribute Specification System

NACO uses a flexible, JSON-based system for defining attribute requirements. Specifications are organised into *attribute categories*, each representing a semantic concept (e.g., "Identifier", "Name", "E-Mail"). Within each category, multiple *allowed keys* define the acceptable attribute names and their expected properties.

The basic attribute set (mandatory) defines the core attributes that every Community AAI should provide:

- Identifier: a non-reassignable, persistent, unique user identifier. Acceptable keys include the compound attribute `sub+iss` (combining OIDC subject and issuer), `eduperson_unique_id`, or `voperson_id`.
- Name: user's display name, accepted via the `name` claim.
- E-Mail: contact email address, via `email` or `voperson_verified_email`.
- Affiliation: home organisation affiliation via `voperson_external_affiliation`, with external scope validation.
- Assurance: identity assurance levels via `eduperson_assurance` or equivalent claims.

The extended attribute set (optional) includes additional attributes that enhance functionality:

- Entitlements and capabilities (`eduperson_entitlement`)

PoS(ISGC2026)032

- ORCID identifier (`orcid`)
- SSH public keys (`ssh_public_key`)
- Policy agreements (`voperson_policy_agreement`)
- Supplemental name information (`given_name`, `family_name`)
- Home organisation domain (`schac_home_organization`)

A key design principle is *flexible matching*: for each attribute category, NACO accepts any of the defined allowed keys. For example, if a Community AAI provides `name` but not `display_name`, the Name category is satisfied. This approach accommodates the variation inherent in federated systems while still ensuring that the semantic requirement is met.

3.3 Validation Logic

NACO performs validation across multiple OIDC data sources: the UserInfo endpoint (the standard OIDC endpoint for retrieving user claims), the Access Token body (for JWT-formatted access tokens, claims embedded in the token payload), and Token Introspection (claims returned by the introspection endpoint, which requires client credentials).

For each attribute category and each source, NACO performs several checks. Presence checking verifies that at least one of the allowed keys is present; for compound attributes (e.g., `sub+iss`), all component keys must be present simultaneously. Type validation checks that attribute values match expected types, supporting `str`, `list`, `bool`, and `dict` types, with a flexible `string_or_list` option for attributes that may appear in either format. Scope validation applies to scoped attributes (those containing an `@` separator), ensuring that external scopes (from home IdPs) match configured allowed domains, while community scopes must not include external domains.

The validation produces a structured verdict containing per-attribute results indicating presence, type correctness, and scope validity; a per-source breakdown showing which endpoints provide which attributes; an overall summary verdict for the Community AAI; and timestamps for tracking when validation occurred.

3.4 Token Handling

Continuous monitoring requires automated access to OIDC tokens from each monitored Community AAI. NACO supports two token acquisition mechanisms. The mytoken service [11] is a token delegation service that issues long-lived JWT tokens (mytokens) which can be exchanged for short-lived access tokens; NACO stores mytokens in its configuration and automatically retrieves fresh access tokens as needed. Alternatively, oidc-agent [12] is a desktop-based token management agent that handles OAuth flows and token refresh, and NACO can retrieve tokens from named oidc-agent accounts.

For token introspection and UserInfo retrieval, NACO integrates with the `flaat` library [10], which provides a unified interface to OIDC operations across different providers.

PoS(ISGC2026)032

4. Implementation

4.1 Continuous Monitoring

NACO is designed for continuous, automated monitoring of Community AAI attribute compliance. The command-line tool can be executed periodically via cron, typically on an hourly basis. Each run retrieves fresh tokens from all configured Community AAIs, validates them against the attribute specification, and updates the cached results. This continuous approach enables early detection of configuration drift or upstream changes before users experience service disruptions.

4.2 Web Service API

The `naco-web` service exposes a REST API that enables integration with external systems. The `POST /analyse` endpoint accepts OIDC token data or SAML assertions for validation, including authentication credentials, target Community AAI identifier, and optional TTL settings. The `GET /list_entries` endpoint returns a list of all cached Community AAI entries, while `GET /get_entry/{op_name}` retrieves cached validation results for a specific Community AAI.

This API has been integrated with the CAT monitoring infrastructure used by EGI [13], allowing attribute compliance to be tracked alongside other service health metrics.

4.3 Caching and Result Persistence

NACO uses a SQLite-based disk cache [9] for persistent storage of validation results. Each cache entry has a configurable time-to-live (TTL), with a default of 16 days. This allows results to remain available even if token retrieval temporarily fails, while ensuring that stale data is eventually purged. The TTL can be overridden on a per-entry basis through the API.

The cache stores both validation verdicts and the raw token data used to produce them, enabling debugging and historical analysis. Entries are timestamped to track when validation last occurred.

4.4 Web Interface and Reporting

Validation results are presented through a web interface that displays structured tables showing each attribute category and its status across validation sources. The interface uses colour-coded indicators (green for compliant, red for missing or invalid) and separates mandatory from optional attributes. Summary statistics and last-update timestamps provide at-a-glance status information.

An email notification system alerts administrators when validation fails or when mytoken credentials expire, including diagnostic information to facilitate troubleshooting.

5. Deployment and Results

5.1 Monitored Community AAIs

NACO is currently deployed to monitor ten different Community AAIs across European and international research infrastructures. The European research infrastructure AAIs include EGI Check-in, eduTEAMS (GÉANT's collaboration platform), LifeScience AAI, and Unity (the Helmholtz Association's identity infrastructure). German research infrastructure is represented by RegAPP,

PoS(ISGC2026)032

Bildungsproxy, academic-id, and didmos. International collaborations include WLCG IAM (serving high-energy physics) and CILogon (serving the US research community).

Validation results are published at a public dashboard, providing transparency into the attribute compliance status of each monitored infrastructure.

5.2 Use Cases

NACO supports several operational use cases. For compliance monitoring, periodic validation of all configured Community AAIs enables early detection of configuration drift or upstream changes; when a previously compliant AAI starts failing validation, administrators are alerted before users experience service disruptions.

For service provider onboarding, NACO can verify that a Community AAI meets attribute requirements before integration, preventing integration failures and identifying gaps that need to be addressed. In federation operations, NACO enables side-by-side comparison of attribute coverage across Community AAIs, supporting standardisation efforts by making the current state of interoperability visible and measurable.

Finally, as AARC-G056 and other specifications evolve, NACO helps validate that proposed requirements are achievable by checking current AAI implementations against draft specifications.

5.3 Findings

Deployment of NACO has revealed several common attribute compliance issues. Not all Community AAIs provide the preferred `voperson_id` attribute; some rely solely on the compound `sub+iss` identifier, which while unique, is less portable across protocol translations. The assurance attribute appears under different names (`eduperson_assurance`, `assurance`, `ass`) and with varying value formats across AAIs. Some scoped attributes include incorrect scope suffixes, typically due to misconfiguration in attribute release policies. While most AAIs provide the mandatory basic attribute set, coverage of optional attributes (ORCID, SSH keys, policy agreements) varies significantly.

These findings have been shared with Community AAI operators, leading to configuration improvements and increased awareness of interoperability requirements. The visibility provided by NACO's public dashboard has created positive pressure toward compliance.

6. Future Work

Several directions for future development are planned. We aim to monitor additional Community AAIs, particularly those serving research communities in Asia-Pacific and other regions, as global interoperability requires global monitoring. Once AARC-G056 is finalised, NACO will be updated to ensure full compliance with the official requirements.

While NACO currently monitors Community AAIs, the root cause of many compliance issues lies with home IdPs that do not release required attributes. A planned feature is to aggregate and expose statistics about home IdP attribute release quality (in anonymised form), creating visibility into upstream problems.

PoS(ISGC2026)032

Adding trend analysis capabilities would enable tracking of compliance over time, identifying patterns of degradation, and measuring the impact of configuration changes. Finally, we plan integration with monitoring platforms beyond CAT of EGI, and development of dedicated compliance dashboards with alerting and SLA tracking capabilities.

7. Conclusion

Interoperability in federated research infrastructure depends on consistent attribute release by Community AAIs. Without automated tooling, compliance verification is sporadic and reactive, allowing problems to persist until users encounter service failures.

NACO addresses this gap by providing continuous, automated validation of attribute compliance against specifications such as AARC-G056. Its flexible specification system accommodates the inherent variation in federated systems while still ensuring that semantic requirements are met. The dual CLI and REST API interfaces support both operational monitoring and integration with external systems.

Deployment across ten European and international Community AAIs has demonstrated NACO's effectiveness in detecting compliance issues and driving improvements. The visibility provided by public compliance dashboards creates accountability and supports standardisation efforts.

We invite additional Community AAIs to participate in NACO monitoring. Broader coverage will strengthen the federated research infrastructure by ensuring that services can rely on consistent attributes regardless of which Community AAI their users come from. Interested operators are encouraged to contact the authors to arrange integration.

References

[1] AARC-G056 - AARC Profile for expressing identity attributes. (To be published.) https://zenodo.org/doi/10.5281/zenodo.23340193

[2] AARC Blueprint Architecture 2019. *Zenodo*, 2020. https://zenodo.org/doi/10.5281/zenodo.3672784

[3] OpenID Foundation. OpenID Connect Core 1.0. https://openid.net/specs/openid-connect-core-1_0.html

[4] AARC-G031: Guidelines for the evaluation and combination of the assurance of external identities. https://aarc-community.org/guidelines/aarc-g031/

[5] AARC-G025: Guidelines for expressing affiliation information. https://aarc-community.org/guidelines/aarc-g025/

[6] AARC-G069: Guidelines for expressing group membership and role information. https://aarc-community.org/guidelines/aarc-g069/

[7] ORCID – Open Researcher and Contributor ID. https://orcid.org/

PoS(ISGC2026)032

[8] Ramírez, S. FastAPI – modern web framework for building APIs with Python. https://fastapi.tiangolo.com/ (Visited: Tue, 13 Oct 2026)

[9] SQLite – a small, fast, self-contained SQL database engine. https://www.sqlite.org (Visited: Tue, 13 Oct 2026)

[10] flaat – FLexible Authentication and Authorization Tool for OIDC. https://github.com/indigo-dc/flaat (Visited: Tue, 13 Oct 2026)

[11] mytoken – Refresh Token Delegation https://mytoken-docs.data.kit.edu (Visited: Tue, 13 Oct 2026)

[12] Zachmann, G. and Hardt, M. and Gudu, D. oidc-agent - Integrating OpenID Connect Tokens with the Command Line. Comput Softw Big Sci 9, 8 (2025). https://doi.org/10.1007/s41781-025-00137-4

[13] Pedersen, A.N. and Sjöström, A. and Thermolia, C. and Groep, D. and Gudu, D. and Georgilakis, K. and Hardt, M. and van Es, M. and Liampotis, N. and Machalet, P. and Lényi, P. and Thijsse, P. and Zamani, T. AARC Community / GRNET. AARC Compliance Assessment Toolkit (CAT). https://aarc3.cat.argo.grnet.gr (accessed 13 October 2026).

PoS(ISGC2026)032