# Development and the Deployment of the Startup System for the ASTRI Mini-Array Project

I. Abu,[a] F. Gianotti,[a] A. Bulgarelli,[a] N. Sayes,[i] H. Campos,[i] M. Araya,[i]
E. Borciani,[a] P. Bruno,[b] P. Calabretto,[a] L. Castaldini,[a] I. Huerta,[i] V. Conforti,[a]
M. Corpora,[e] A. Costa,[b] A. Di Piano,[a] F. Fiordoliva,[d] V. Fioretti,[a] S. Gallozzi,[d]
A. Giuliani,[f] A. Grillo,[b] M. Lodi,[h] F. Lucarelli,[d] A. Mandica,[a] G. Mattioli,[a]
K. Munari,[b] N. Parmiggiani,[a] G. Pareschi,[c] V. Pastore,[a] F. Russo,[a] S. Scuderi,[c]
A. Tacchini,[a] G. Tosti,[g] S. Spinello[b] for the ASTRI Project[j]

[a]INAF - Osservatorio di Astrofisica e Scienza dello Spazio, Bologna, Italy
ismam.abu@inaf.it

[b]INAF - Osservatorio Astrofisico di Catania, Via S. Sofia 78, Catania, Italy

[c]INAF - Osservatorio Astronomico di Brera, Via Bianchi 46, Merate, Italy

[d]INAF - Osservatorio Astronomico di Roma, Via Frascati 33, Roma, Italy

[e]IASF - Istituto di Astrofisica Spaziale e Fisica Cosmica di Palermo, Italy

[f]IASF - Istituto di Astrofisica Spaziale e Fisica Cosmica di Milano, Italy

[g]Università di Perugia, Dip. di Fisica e Geologia, Via A. Pascoli, Perugia, Italy

[h]Fundación Galileo Galilei – INAF, Santa Cruz de Gran Canaria, Spain

[i]Universidad Técnica Federico Santa María, Gral. Bari 699, Valparaíso, Chile

[j]http://www.astri.inaf.it/index.php/new-general-info/
new-astri-team/

**Abstract.** The Startup System of the ASTRI Mini-Array is a portable, containerized software that orchestrates startup and shutdown procedures and provides real-time monitoring across heterogeneous subsystems. A System Configuration Database (SCDB, MongoDB) describes the infrastructure and drives automated initialization of the telemetry and alerting backend. Admin-Tools derives from SCDB the InfluxDB layout (buckets, tasks, and conventions), while protocol-specific collectors (SNMP, OPC UA, Modbus, and Telegraf) stream measurements into InfluxDB. A lightweight webserver exposes selected streams to a operator interface and to alarm services. Development followed an incremental path from fully simulated telemetry to progressive integration of real interfaces, and deployment moved from a virtualized testbed to on-site hosts and on-sky validation with ASTRI-3. On production servers, the end-to-end stack reaches an operational ready state in about five minutes, enabling reliable recovery and consistent operations. The resulting SCDB+InfluxDB backbone is also suitable as a data foundation for future predictive maintenance and digital-twin applications.

## 1.   Introduction

In this paper we present the Startup System, the software system responsible for managing the startup and shutdown sequence and the critical monitoring of all on-site ASTRI Mini-Array subsystems. These subsystems include Power, Environmental, Timing and ICT, Safety and Security, Service Cabinets, the Telescopes, and the SCADA system.

The ASTRI Mini-Array is an international project led by INAF. It will operate nine dual-mirror imaging atmospheric Cherenkov telescopes at the Teide Observatory in Tenerife, described in detail by Scuderi et al. (2022) and Lombardi et al. (2022b).

The array targets very-high-energy gamma-ray sources. It also enables stellar intensity interferometry, as described by Vercellone et al. (2022), Zampieri et al. (2022) and Lombardi et al. (2022a).

The observatory subsystems are coordinated by a SCADA framework, described in detail by Bulgarelli et al. (2024) and runs in the ICT System, described in detail by Gianotti et al. (2024), there is also a remote installation, in the TestBed of ASTRI Mini-Array, hosted in the INAF-OAS Bologna's Datacenter, described in detail by Gianotti et al. (2022).

The Startup System automate the operation of the SCADA Software, and it enables non-expert operators to carry out observations. Without orchestration, operators would need to manually manage dozens of interdependent services, a process prone to errors and unsafe conditions. The Startup System ensures that all critical components are started in the correct order, dependencies are respected, and the system reaches a consistent ready state.

The architecture of the Startup System is described in detail by Bulgarelli et al. (2025). This paper describes both the conceptual model and the practical path taken to realize it. We highlight the orchestration principles, the incremental development process and the results of the first deployments, including on-sky validation.

## 2.   System Overview

The Startup System orchestrates startup and shutdown sequences and performs continuous readiness and health checks for the main on-site subsystems, including Power, Environmental, Timing and ICT, Safety and Security, Service Cabinets, Telescopes, and SCADA components. The design goal is portability: the same stack can be deployed on the on-site ICT server, an operator laptop for emergency recovery, or a remote instance hosted in the Bologna testbed and connected to the observatory through a VPN.

**Deployment** is intentionally simple and requires only Docker. The stack is defined as a Docker Compose application and is started through a single command interface that selects execution modes. These modes support fully simulated runs, mixed configurations, and full operation with real telemetry. The same interface provides stop procedures that either preserve databases for operational continuity or reset parts of the environment for testing and development.

**Configuration** is centralized in the System Configuration Database (SCDB), which provides a machine-readable description of devices, endpoints, and conventions. From SCDB, Admin-Tools generates the InfluxDB initialization (buckets, retention policies, tasks for shaping data, and naming and tagging conventions) and exports validated configuration fragments for collectors and dashboards. Collectors ingest telemetry using the relevant protocols: Telegraf agents cover host-level metrics, while dedicated ser-

vices interrogate SNMP devices, OPC UA servers, and Modbus endpoints. Measurements are stored as time series in InfluxDB, while alarms are collected and forwarded to the operator interface.

**Readiness** is defined by a small set of checks: service health, reachability of SCDB and InfluxDB, and the presence and freshness of key telemetry streams. Freshness checks prevent stale data from being interpreted as current status. Operational procedures are idempotent: components that are already healthy are not restarted, which reduces downtime and supports safe repeated execution during recovery and maintenance.

## 3. Development and Deployment

Implementation progressed in small iterations, starting from a minimal core that deployed SCDB and InfluxDB and produced synthetic telemetry. This stage validated the end-to-end data path and established the conventions used by dashboards and collectors. After stabilizing the telemetry pipeline, we introduced the real-time relay to feed the operator interface and implemented the first dashboards.

The next milestone was the alarm subsystem, which enabled fault reporting and supported operator workflows. Only after this software baseline was stable, we moved to integration with real subsystem interfaces. Protocol-specific collectors were added progressively and tested first in the Bologna testbed, which mirrors the observatory ICT environment and allows integration tests without risks for on-site operations.

As real telemetry sources were integrated, we addressed signal volume and operator usability by shaping data at the database level. InfluxDB tasks are used to filter, aggregate, and derive operational indicators while keeping raw streams available for engineering analyses. This approach reduces code churn because changes in the displayed or alarmed signals can be implemented by updating tasks rather than modifying collectors.

Deployment followed three steps: (i) testbed validation of the conceptual model, (ii) on-site installation with simulated telemetry to verify packaging and runtime behavior, and (iii) on-site operation with real telemetry and SCADA interactions to support observation procedures.

## 4. Validation and Test

Validation combined integration testing, fault injection, and performance measurements. Functional tests verified that the system reaches a ready state when the defined checks are satisfied and that repeated startup or shutdown executions are safe due to idempotent procedures.

Fault scenarios were exercised by simulating failures along subsystem chains and by inducing resource and network degradation. In these conditions, dashboards highlighted degraded components and alarms were produced with meaningful severity, enabling operators to identify and react to problems quickly.

Performance was measured on representative hosts. A full installation from scratch reaches an operational ready state in about five minutes on a production-class server, and within about eight to ten minutes on a laptop-class machine. Finally, the stack was validated during real observation tests, where it operated with real telemetry and interacted correctly with SCADA components during on-sky activities with ASTRI-3.

## 5.    Conclusions and future work

The Startup System provides a portable and easy-to-install software solution that automates startup and shutdown operations and supports continuous monitoring for the ASTRI Mini-Array Project. The deployment strategy is flexible and includes a standard on-site server, an operator laptop for emergency control, and a remote recovery installation hosted in Bologna and connected to the observatory through a dedicated VPN.

The system relies on Docker as its only external dependency and uses a Docker Compose stack that can be started through a dedicated run command supporting multiple execution modes. The architecture combines telemetry ingestion into InfluxDB, alarm collection through a dedicated alarm server, and a webhook-based delivery mechanism that keeps the frontend updated. A key element is the System Configuration Database, which stores a detailed description of the observatory and enables consistent behavior across collectors, dashboards, and orchestration logic. Readiness checks and safe repeated execution of procedures improve reliability and operational safety, and they also simplify development and maintenance.

A major future direction is the use of SCDB and InfluxDB as the base for a Digital Twin. SCDB describes the structure and configuration of the system, while InfluxDB provides the time series history of subsystem behavior. Together they can support simulation, prediction, and what-if analyses. This approach is also relevant for future adoption in the SST context for CTAO, where a similar orchestration and monitoring layer could benefit from the same data-driven foundation.

## References

Bulgarelli, A., Abu, I., & et al. 2025, in Proceedings of ADASS, vol. This Conference
Bulgarelli, A., Lucarelli, F., Tosti, G., Conforti, V., Parmiggiani, N., Schwarz, J. H., Araya, M., & et al. 2024, Journal of Astronomical Telescopes, Instruments, and Systems, 10
Gianotti, F., Abu, I., Lodi, M., Tacchini, A., & et al. 2024, in Proceedings of SPIE, vol. 13101
Gianotti, F., Conforti, V., Tacchini, A., Bulgarelli, A., Bruno, P., & et al. 2022, in Astronomical Data Analysis Software and Systems XXX, ASP Conference Series, vol. 532
Lombardi, S., Lucarelli, F., Bigongiari, C., & et al. 2022a, in Proceedings of SPIE, vol. 12189
Lombardi, S., Scuderi, S., Gargano, C., Giordano, V., Giuliani, A., Grivel, C., La Palombara, N., Lessio, L., Millul, R., & et al. 2022b, in Proceedings of SPIE, vol. 12189
Scuderi, S., Giuliani, A., Pareschi, G., Tosti, G., Catalano, O., Amato, E., Antonelli, L. A., Becerra Gonzáles, J., & et al. 2022, Journal of High Energy Astrophysics, 35
Vercellone, S., Bigongiari, A., Burtvoi, M., Cardillo, M., Catalano, O., & et al. 2022, in Journal of High Energy Astrophysics, Volume 35, August 2022, Pages 1-42, vol. 12189
Zampieri, L., Bonanno, G., Bruno, P., & et al. 2022, in Proceedings of SPIE, vol. 12183