The Italian National Institute for Astrophysics (INAF) manages three radio telescopes: the Medicina and Noto dishes and the newly-built SRT [1]. In order to make their capabilities more valuable to the scientific community, we started the DISCOS (Development of the Italian Single-dish Control System) project. DISCOS is implemented according to a distributed Component-Container model provided by the ALMA Common Software (ACS) framework [2], and it hides to the users the differences among the telescopes by presenting the same user interface and the same data format.

The complexity of coping with three heterogeneous instruments was handled by designing a software development infrastructure with a wide monolithic codebase (libraries, components and generic interfaces), which is completely shared among the three product lines while specific differences are managed via configuration files or via polymorphism only when strictly necessary. This design permits to produce new software components with a minimum effort and to set up the same development and maintenance process for all the environments, thus leading to an affordable development and maintenance process.

**WORKFLOW**

In recent years we have tried to formalize the development workflow of new components, resulting in cleaner and more maintainable code.

- **development** of new components is split into: development of a hardware simulation server, development of a standalone hardware communication library and development of the ACS component exploiting the library
- **maintenance** is handled with regression tests: each time a bug is found, the developer writes a test that reproduces the buggy behaviour, then the fix is committed into the patch release until the test executes correctly.
- **continuous integration** is performed via a Jenkins server. At present we run nightly builds of the trunk branch of each telescope and of every maintained release. This infrastructure also permits test automation and to collect the results on the whole project.

**DESIGN**

The design of the DISCOS control software highly relies on the ACS patterns and services. In the ACS model, the basic unit performing a task is a component, each component exposes an interface and is individually configured to determine its exact behaviour inside the system. The station-specific modules consist essentially in the low-level and no-logical control of the devices and of the telescope hardware.

The complexity of coping with three heterogeneous instruments was handled by designing a software development infrastructure with a wide monolithic codebase (libraries, components and generic interfaces), which is completely shared among the three product lines while specific differences are managed via configuration files or via polymorphism only when strictly necessary. This design permits to produce new software components with a minimum effort and to set up the same test suites for all the environments, thus leading to an affordable development and maintenance process.

**Table:** basic features of the three antennas. Right (top to bottom): schematic of the control system architecture.

The DISCOS project documentation can be found following the above QR code.