Part I of this manual describes how to write a monitor. It discusses the features and syntax of the SMEDL specification language. Part II is about integrating monitors into a larger system. It describes the API for the generated code and the various transport mechanisms available for asynchronous connections.

Part I.

The SMEDL and A4SMEDL Monitoring Language

A SMEDL monitoring system is designed at two levels: the monitor level and the architecture level. Monitor specifications describe a single monitor, that is, a list of state variables, a list of events that the monitor consumes and emits (known as "imported" and "exported" events), and a set of states, transitions, and actions describing the state machine. Architecture specifications describe how multiple monitors come together to form a monitoring system, including which monitor specifications are involved, how instances of the monitors are parameterized, and how imported and exported events are directed.

Monitor specifications are written in `.smedl` files, and there may be multiple in a monitoring system. Architecture specifications are written in `.a4smedl` files and there is exaclty one per monitoring system.

1. SMEDL Specifications

A SMEDL specification describes a single monitor. This is where the actual monitoring logic is definied. At its core, a SMEDL monitor is a set of state machines. Transitions are triggered by events, with optional conditions and associated actions. Monitors may also contain state variables, which the conditions and actions can make use of.

2. A4SMEDL Specifications

Part II.

The SMEDL Programming Interface

3. Common Elements

4. Monitor API

5. Local Wrapper API

6. Global Wrapper API

7. Transport Adapters

7.1. RabbitMQ

7.2. File