



Fig. 1 A Service Monitoring Architecture

level, the PAC escalates the issue to a higher level component, namely the POC. In case of an SLA violation, adjustment can trigger re-planning, re-configuration and/or alerting to higher-level SLA monitoring. These capabilities are considered to be important in order to assure preserving service provision and resource quality.

The MonitoringManager (MM) coordinates the generation of a monitoring configuration of the system. It decides, for an SLA specification instance it receives, which is the most appropriate monitoring configuration according to configurable selection criteria. A monitoring configuration describes which components to configure and how their configurations can be used to obtain results of monitoring Guarantee States. The Low Level Monitoring Manager is a central entity for storing and processing monitoring data. It collects raw observations, processes them, computes derived metrics, evaluates the rules, stores the history and offers all this data to other components (accessible through the ServiceManager). It implements the monitoring part of a ProvisioningRequest, containing constraint based rules (time and data driven evaluations) and ServiceInstance specific Sensor related configurations. It is general by design, so capable of supporting monitoring of software services, infrastructure services and other resources.

There are three types of Monitoring Features in the monitoring system. First, *Sensors* collect information from a service instance. Their designs and implementations are domain-specific. A sensor can be injected into the service instance (e.g., service instrumentation), or it can be outside the service instance intercepting service operation invocations. A sensor can send the collected information to a communication infrastructure (e.g. an Event Bus) or other components can request (query) information from it. There can be many types of sensors, depending on the type of information they want to collect, but all of them implement a common sensor interface. The interface provides methods for starting, stopping, and configuring a sensor. Second, *Effectors* are components for configuring service instance behaviour. Their