In DGP, logging is focused on capturing data from events that occur during the use of a system. Together with monitoring, they provide observability into the state of each tier in a DGP system, while other tools are used to collect lower-level data from host operating systems, app servers, networks, etc. The consolidation of all logic into the web services simplifies logging, which only has to accommodate client applications and server-tier web services.

Requirements

1. Consistent and reusable logic to capture event data from all the tiers of a system.

What: *Data from events that occur in the N-tier distributed system must be captured locally to a log, and should ideally also be written to a centralized database.*

Why: *Writing the event data to a local log is the most reliable mechanism, and acts as a backup for the centralized log database. Writing the event data to a centralized database makes notifications easier, and is also frequently used for operations dashboards, reports and analysis.*

Testing: *The capture of event data should be implemented as API methods, and use API Tester test files to verify their correct functionality in each environment.*

2. Admin notification of events.

What: *System administrators (operations) must be notified when serious events occur in a system.*

Why: *Tools such as Splunk or an equivalent must be used to monitor the event logs and databases for serious system events that require admins to be notified.*

Testing: *The tools to scan event data and notify admins can be tested in the test and QA environments.*