Correlation Business Logic 1

- OpenNMS is enhanced with rule based correlation (see JSR-94) and the user is able to change alarm, and notification behaviors based on business rules

- Alarm Correlation capability – user benefit
  - Ability to connect to external correlation capability which can provide support to human decision making
    - Remove ‘wall of red’
    - Auto acknowledge sympathetic alarms
    - Identify route cause alarms
    - Change alarm prioritizations based upon business logic (time of day / configuration etc)
  - BASIC: Linear alarm list processing (boolean logic)
  - ADVANCED: Topology based alarm processing – process alarms against common topology model
  - Ability to tag correlated problems to trouble ticket for management
    - (Notification is like TT but single event)
Correlation / Topology / Trouble Ticket

TT App

Forward TT to TT application

Automatic generated trouble with attached alarms

Manually generated trouble with attached alarms

Corelation

Business Rules

Topology Inventory

Alarm Management

OpenNMS

Alarm List

Event Processing

Topology Inventory

discovery

SNMP events

NW

Align Correlation Inventory to OpenNMS inventory
Correlation Business Logic 2

• Base case:
  - ‘OpenNMS is enhanced with rule based correlation (see JSR-94) and the user is able to outage, alarm, and notification behaviors based on business rules’

• Alarm Correlation Interface
  - API to allow separate alarm correlation Module to connect to OpenNMS alarm list and process alarms using rules db. This allows choice of engines and for injection of new/modified/ correlated alarms from external systems engines

• Topology Interface
  - Ability to export and / or align opennms topology / inventory database with external rules and/or topology model system

• TTicket Interface
  - Ability to ( manually / and / or automatically ) generate trouble tickets with tagged route cause and sympathetic alarms and user comments. Ability to forward this to external TT system
Corellation Business Logic 3

- Alarm Processing – base bahaviour
  - Alarm Cycle
    - New alarm -> new alarm acknowledged
    - alarm cleared -> alarm cleared & acknowledged
    - alarm cleared and acknowledged -> archive
  - Mark incoming alarms which are root cause alarms
  - Create new route cause alarms from sympathetic alarms
  - Mark sympathetic alarms as acknowledged and correlated to route cause
  - Support OSS alarm processing paradigm
    - In particular Managed object instance and managed object type and application DN/AlarmID
    - Support the X733 alarm definition.
Corellation Business Logic 4

• Gui
  - Filter / display by route cause alarms and corellated sympathetic alarms (auto acknowledge corellated?)

• Corellation Engine – base technology
  - Ideally based upon JSR93 corelation interface – Jboss Drools, JESS, ILOG Jrules
  - Easy to write business rules – rule writing tool kit

• Trouble Ticket interface
  - (Trouble Ticket I/F) attach route cause and sympathetic alarms to trouble ticket (Trouble ticket interface)
  - (Trouble Ticket I/F) Manually tag alarms to attach to trouble ticket
Corellation Business Logic 5 - Enablers

- OSSJ alarm interface (X733)
  - JVT (ejb) alarm interface – events and queries
  - XVT (XML) alarm interface - events and queries

- Base use cases
  - OpenNMS can connect to another OSS/J interface and:
    - Register for alarm events (new, cleared, acknowledge, change, comments?)
    - Resynchronize with current alarm list from remote system
    - Receive new alarms events from other system
    - Receive acknowledge events
  - Other system can connect to OpenNMS OSS/J interface and:
    - Register for alarm events (new, cleared, acknowledge, change, comments?)
    - Resynchronize the alarm list from openNMS

- OSSJ TT interface