

LMAP 2016-02-22 Interim Input for Discussion

Jürgen Schönwälder, Jacobs University

Granularity ma-periodic-interval

- Issue:
 - Sub-second granularity for starting tasks is expensive to implement and generally not needed
- Proposal:
 - Change unit from milliseconds to seconds
 - Leave ma-timing-random-spread at millisecond resolution

Glob-style tag matching

- Issue:
 - Selection of suppression targets is either all (*) or a list of explicit matches. This provides rather limited flexibility (e.g., it is not possible to match all schedules tagged with fcc-*).
- Proposal:
 - Rename ma-suppression-tags to ma-suppression-match (fixing a misnomer)
 - Allow glob style pattern: *, ? and [] ranges, essentially matching POSIX fnmatch() behavior

Streamline tags and suppression tags

- Issue:
 - Support tags and suppression tags on tasks and schedules and actions. Tags are echoed with the result records, suppression tags are not echoed.
- Proposal:
 - Add tag sets where they are missing
 - Remove ma-report-task-cycle-id and ma-task-cycle-id, tags are a more general replacement

Remove ma-task-suppress-by-default

- Issue:
 - Definition is incomplete how exactly default suppression is invoked.
 - The tagging mechanism makes it easy to define the a default suppression policy
- Proposal:
 - Remove ma-task-suppress-by-default, tasks can be easily tagged to emulate ma-task-suppress-by-default

Overlapping schedules / actions

- Issue:
 - Executing an action A while a previously started action A has not terminated yet can quickly lead to resource outages
- Proposal:
 - Add text that says the execution of actions or schedules is skipped if previous invocations are still running
 - Add a 'running' state to schedules and actions

Add counters to ma-status-action-obj

- Issue:
 - There are no counters indicating how frequently actions are invoked, how often they have terminated successfully, and how often they have terminated with a failure
 - Counters cost almost nothing but help to easily spot non-spurious errors
- Proposal:
 - Add suitable counters: # invocations, # failures, # of suppressed invocations, # of overlapping invocations, ... to ma-status-action-obj

Identifiers for surrogate keys and tags

- Issue:
 - Surrogate keys and tags are essentially strings. This allows interesting keys and tags such as ‘../../foo’ or ‘> /dev/null’ that require implementations to take special care to avoid injection attacks
- Proposal:
 - Introduce a type for surrogate keys and tags that allows only a combination of letters, hyphens, underscores and digits (e.g. yang:yang-identifier) [question also posted to yang-doctors]
 - Alternative: continue using strings but add explicit text into the security considerations?

Calendar semantics clarification

- Issue:
 - Is a calendar schedule with `ma-calendar-start == ma-calendar-end` and all other fields set to wildcards going to fire or not?
- Proposal:
 - The question is whether `ma-calendar-start` and `ma-calendar-end` are inclusive. I think they should be both inclusive, which means, the above will never trigger.

More flexible calendar schedules?

- Issue:
 - Calendar event expressions are relatively basic: no support for ranges, no support for ranges with steps
- Proposal:
 - Allow more powerful expressions:

`<minute>0,15,30,45</minute>`
`<minute>0-59/15</minute>`
`<minute>*/15</minute>`

`<month>april-september</month>`
 - This may turn sets into scalars in the information model (and leaf-lists into leafs in the YANG model).
 - Support these as a feature in the YANG model?

Role of ma-task-obj

- Issue:
 - An implementation needs to be configured which tasks are supported etc. The ma-task-obj pretty much serves this purpose.
- Proposal:
 - The ma-task-obj is ‘internal’ configuration and typically not exposed to controllers for configuration (that is, ma-task-obj is not part of what the information model calls instructions).
 - In the YANG model, mark the ma-task-obj leafs with `nacm:default-deny-write`