

**IBM Tivoli Netcool**

**Parent Child Relationship Management**

**Innovise ESM**

**Keypoint**

**High Street**

**Slough**

**SL1 1DY**

**Tel: +44 (0) 1753 513 800**

**Author: Chris Janes**

**Title: Parent Child Relationship Management**

**Policies**

**Version: 0.00**

Contents

[1. Document Control 3](#_Toc266086901)

[2. Introduction 4](#_Toc266086902)

[3. General Policies 4](#_Toc266086903)

[Appendix 1 – Scripts used to create the objects required 5](#_Toc266086904)

[Create Parent Child Table 5](#_Toc266086905)

[Create Parent Child Filter Table 5](#_Toc266086906)

**List of Figures**

**No table of figures entries found.**

**List of** **Tables**

**No table of figures entries found.**

# Document Control

**Document Prepared By:**

The following Innovise Limited personnel have prepared this document:

**Name Title**

Chris Janes Consultant

**Document Reviewed By:**

The following Innovise personnel have reviewed this document:

**Name Title**

|  |  |
| --- | --- |
| **Name** | **Title** |
|  |  |
|  |  |
|  |  |

**Document Revision History:**

The following versions have been distributed:

**Version Revised and Issued By:**

0.0 Chris Janes

**Number of Copies Submitted to Customer:** 1

**Number of Copies for Innovise Limited:** 1

**Agreed and approved on behalf of Customer**

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Title: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_**

# Introduction

This document has been produced describe Parent Child relationship management of events within the Netcool NMS Solution.

There are defined relationships between Events that allow the number of events being presented to the system operators by suppression of Child events. This relationship is held in a Netcool ObjectServer Table (custom.parentChild)

# ObjectServer Tables

## Parent Child Table

Source\_Event\_ID varchar(256) PRIMARY KEY,

This Field contains the EventID of this Relationship, We generate this manually of the form XXX\_nnn where XXX represents an EMS and nnn is a serial number of the relationship from this EMS

Example: RAD\_006

Parent\_Event\_IDs varchar(256),

This field is the Event ID of any Events that may be a parent to this event. This may be none, one or many events. Where there are many Parents the list is comma separated.

Example: 5620SAM\_002,5620SAM\_003,RAD\_006,OV2540\_003

Child\_Event\_IDs varchar(256),

This field is the Event ID of any Events that may be a child to this event. This may be none, one or many events. Where there are many Children the list is comma separated.

Example: SYS\_010,5620SAM\_001,5620SAM\_006,5620SAM\_014,5620SAM\_018

AlarmClass varchar(128),

This is used with manager to identify this event. This may well represent the AlertGroup of the source event.

Example: aipAMAPStatusTrap

Manager varchar(64),

This is used with AlarmClass to Identifier. This field is the manager of the event.

Example: NEC-SMI

Parent\_Filters varchar(512),

This describes the filter(s) that define the Parent Events (see Parent Child Filter Table). Where there are multiple Filters they are comma separated. Where a filter may have multiple elements to a filter then these are colon separated.

Example:

RemoteNode\_Node:RemoteCard\_Card,RemoteNode\_Node:RemoteCard\_Card,Node\_RemoteNode,Node,RemoteNode\_Node:RemoteCard\_Card,Node\_RemoteNode,NodeLikeRemotenode,NodeLikeRemotenode,NodeLikeRemotenode,NodeLikeRemotenode,Node\_RemoteNode

Child\_Filters varchar(512),

This describes the filter(s) that define the Child Events (see Parent Child Filter Table). Where there are multiple Filters the they are comma separated. Where a filter may have multiple elements to a filter then these are colon separated.

Example:

Node\_RemoteNode,Node\_RemoteNode,Node\_RemoteNode,Node\_RemoteNode:Card\_RemoteCard,Node\_RemoteNode:Card\_RemoteCard,Node\_RemoteNode,Node\_RemoteNode,Node\_RemoteNode,Node\_RemoteNode,Node,Node\_RemoteNode,Node\_RemoteNode:Card\_RemoteCard

Enrichment\_Source varchar(64)

This is used to define where remote Enrichment will be sourced.

## Parent Child Filters Table

FilterName varchar(64)

This describes a filter that is used to define the Parent or Child

Example: Node\_RemoteNode

Definition varchar(256)

This is the filter used to define a parent or Child Event.

Example: (Node = \'@RemoteNode\')

It should be noted that in exceptional case it may be possible to hard code special filter. In this case the filter will not be within brackets

## Alerts Status Table

ParentSerial Varchar(1024)

ChildSerial Varchar(1024)

ParentChild Int

# ObjectServer Triggers

## ParentChildEventID

This trigger is used to set the EventID against M anager and AlarmClass from the Parent Child Table

# Impact Policies

# Appendix 1 – Scripts used to create the objects required

## Create Parent Child Table

CREATE TABLE custom.ParentChild PERSISTENT

(

Source\_Event\_ID varchar(256) PRIMARY KEY,

Parent\_Event\_IDs varchar(256),

Child\_Event\_IDs varchar(256),

AlarmClass varchar(128),

Manager varchar(64),

Parent\_Filters varchar(512),

Child\_Filters varchar(512),

Enrichment\_Source varchar(64)

)

## Create Parent Child Filter Table

CREATE TABLE custom.ParentChildFilters PERSISTENT

(

FilterName varchar(64) PRIMARY KEY,

Definition varchar(256)

)