Starting the Journey
NOI 1.6

Reid Minyen
Hybrid Cloud Solutions Architect

Netcool Operations Insight 1.6

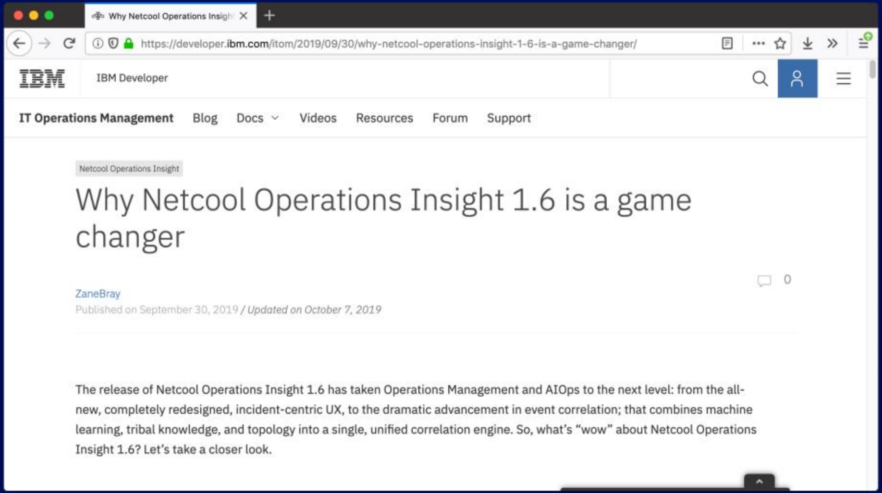
You’ve heard about this…

- New automated analytics
- New correlation engine
- New user experience (UX)

and…

- Deploy on cloud !

But how do you get there?

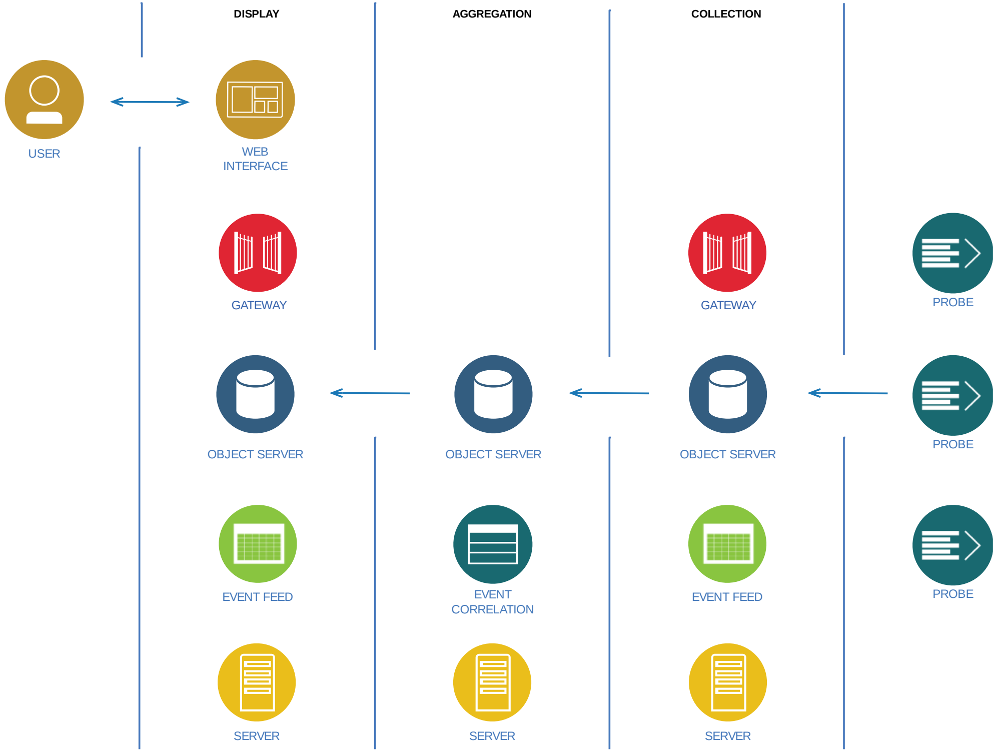

Why Netcool Operations Insigh
https://developer.ibm.com/itom/2019/09/30/why-netcool-operations-insight-1-6-is-a-game-changer/
IBM Developer
IT Operations Management
Blog
Docs
Videos
Resources
Forum
Support
Netcool Operations Insight
0

Why Netcool Operations Insight 1.6 is a game changer

ZaneBray
Published on September 30, 2019 / *Updated on October 7, 2019*

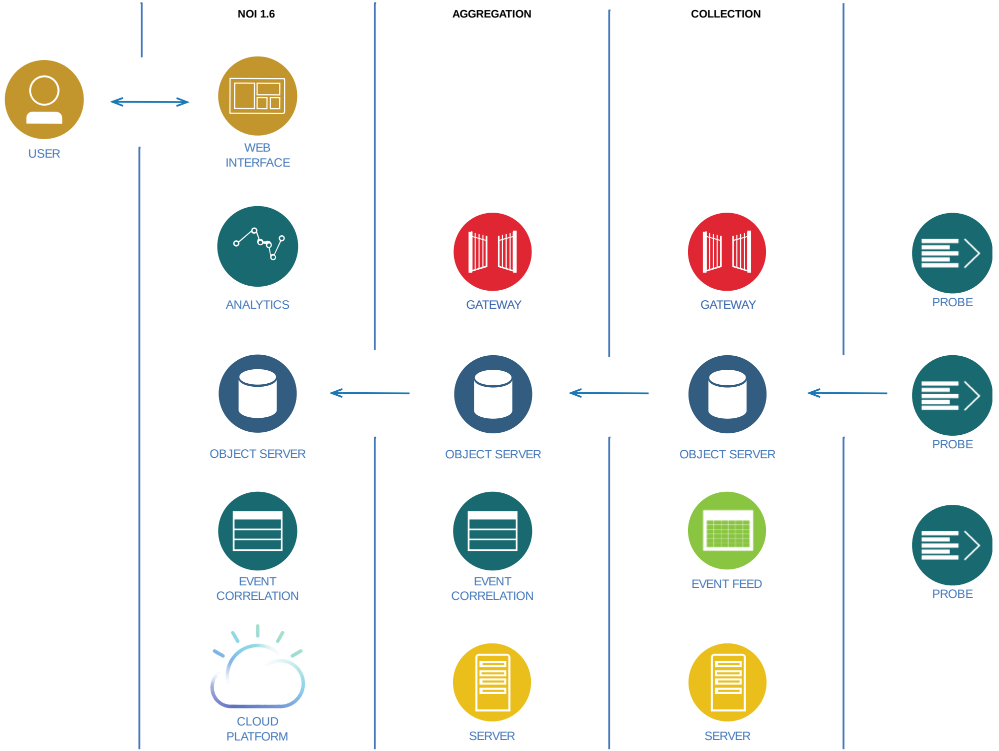
The release of Netcool Operations Insight 1.6 has taken Operations Management and AIOps to the next level: from the all-new, completely redesigned, incident-centric UX, to the dramatic advancement in event correlation; that combines machine learning, tribal knowledge, and topology into a single, unified correlation engine. So, what’s “wow” about Netcool Operations Insight 1.6? Let’s take a closer look.

Here we are… Netcool 3-Tier



Here’s where we can go…

- We deploy 1.6 in any cloud environment (OpenShift, IBM Cloud Private)
- We introduce a unidirectional gateway to the existing AGG synchronizing events over to the new NOI 1.6 “containerized” environment



Simple!   Right?

Actually, we need to cover a few things

1. Create a Gateway.
2. Add entries to your omni.dat for your new 1.6 Object Server and new Gateway. Generate a new interfaces file.
3. Inventory custom fields and custom tables from existing aggregation Object Server.
4. Add custom fields and custom tables to new 1.6 Object Server.
5. Customize the Gateway's mapping.
6. Configure Gateway properties.
7. Configure housekeeping on new 1.6 Object Server.
8. Other tasks required if migrating completely.

This process models the same outlined in…

"**Upgrading IBM Netcool/OMNIbus**
in Production Environments - Best Practices"

Available here: http://ibm.biz/nco_bps

Create a Gateway

Same as it ever was.

The gotcha? Where do you put it?

Place the new gateway somewhere on-premises.

- New box? Install OMNIbus but need only select…
  - ObjectServer gateways
  - Process agent

- Perhaps co-resident with the existing aggregation Object Server itself?
  - Gateway support likely already present.
  - Also present are the multi-tier configuration files, ${OMNIHOME} extensions/multitier/gateway.

Add to omni.dat and interfaces

Know the proxy for the ObjectServer on cloud!

IBM Knowledge Center
- > Netcool Operations Insight 1.6.0
  - > Connecting to event sources
    - > Connecting event sources to your Operations Management on IBM Cloud Private deployment

https://www.ibm.com/support/knowledgecenter/en/SSTPTP_1.6.0/com.ibm.netcool_ops.doc/soc/config/concept/cfg_configuring-event-sources.html

Make sure the FQDN resolves! Add it to /etc/hosts?

Don’t avoid TLS. Practice with it. Get good at it.

Inventory custom fields and tables

- Custom fields must be identified and added to both the new 1.6 NOI ObjectServer and the mapping for the Gateway.
- Custom tables *may* need to be added to the new ObjectServer and Gateway mapping, depending on use.
- nc_osreport is your friend!

IBM Knowledge Center
> Tivoli Netcool/OMNIbus 8.1.0
> Setting up Tivoli Netcool/OMNIbus
> Importing and exporting ObjectServer configurations

https://www.ibm.com/support/knowledgecenter/en/SSSHTQ_8.1.0/com.ibm.netcool_OMNIbus.doc_8.1.0/omnibus/wip/install/task/omn_con_importexportobjservosreport.html

Add custom fields and tables to NOI 1.6 ObjectServer

We use the output of nc_osreport to review what customizations have been made.

If customizations need to be brought over, nco_sql is our friend.

IBM Knowledge Center
- > Tivoli Netcool/OMNIbus 8.1.0
  - > Administering the server components
    - > ObjectServer SQL
      - > SQL interactive interface

https://www.ibm.com/support/knowledgecenter/en/SSSHTQ_8.1.0/com.ibm.netcool_OMNIbus.doc_8.1.0/omnibus/wip/admin/task/omn_adm_sql_usingtxtfilesinput.html

If the number of custom fields to be brought over is small, the SQL for nco_sql could be as simple as just the following…

```
ALTER TABLE alerts.status ADD COLUMN WidgetcomText VARCHAR(255);
go

ALTER TABLE alerts.status ADD COLUMN WidgetcomInt INTEGER;
go

ALTER TABLE alerts.status ADD COLUMN WidgetcomTime UTC;
go
```

Customize the new Gateway’s mapping

We use the **C_TO_A_GATE_P_1** configuration for gateways from…

${OMNIHOME}/extensions/multitier

…as our template.

Copy and modify the .map file for fields, .tablrep.def for tables.

Ensure any replicated custom fields are not set ON INSERT ONLY (as many custom fields are set after insert).

When the number of custom fields brought over is small, additions to the mapping file are also minimal…

```
######################################
#
#       CUSTOM alerts.status
#       FIELD MAPPINGS GO HERE
#
######################################

'WidgetcomText' = '@WidgetcomText',
'WidgetcomInt'  = '@WidgetcomInt',
'WidgetcomTime’ = '@WidgetcomTime',

######################################
```

Customize the Gateway properties

Again, based on…

$OMNIHOME/extensions/multitier/gateway/C_TO_A_GATE_P_1.props

… with a Gateway name of TEMP_GATE, here's a **suggested** set of properties:

```
MaxLogFileSize                      : 2048
MessageLevel                        : 'info'
MessageLog                          : '$OMNIHOME/log/TEMP_GATE.log'
Name                                : 'TEMP_GATE'
Ipc.Timeout                         : 90
Gate.CacheHashTblSize               : 10069
Gate.MapFile                        : '$OMNIHOME/etc/TEMP_GATE.map'
Gate.Mapper.ForwardHistoricDetails  : TRUE
Gate.Mapper.ForwardHistoricJournals : TRUE
```

(continued…)

Customize the Gateway properties (cont.)

```
Gate.StartupCmdFile            : '$OMNIHOME/gates/objserv_uni/objserv_uni.startup.cmd'
Gate.Reader.Server             : 'OLD_PROD_AGG_P'
Gate.Reader.Description        : 'collection_gate'
Gate.Reader.FailbackEnabled    : FALSE
Gate.Reader.IducFlushRate      : 30
Gate.Reader.TblReplicateDefFile : '$OMNIHOME/etc/TEMP_GATE.tblrep.def'
Gate.Reader.IgnoreStatusFilter  : TRUE
Gate.Writer.Server             : 'ICP_AGG_V'
Gate.Writer.Description        : 'temp_gate'
Gate.Writer.BufferSize         : 50
Gate.Writer.FailbackEnabled    : FALSE
Gate.Writer.SAF                : TRUE
Gate.Writer.SAFFile            : '$OMNIHOME/var/objserv_uni/TEMP_GATE.store'
Gate.Writer.UseBulkInsCmd      : TRUE
Gate.Resync.Enable             : TRUE
Gate.Resync.Type               : 'UPDATE'
Gate.Resync.LockType           : 'NONE'
```

Configure housekeeping

Implement housekeeping!

Found in **${OMNIHOME}/extensions/housekeeping**

Prevent accumulating events in the new NOI 1.6 ObjectServer on Cloud.

Over time, de-escalates events before finally retiring them.

Other things to migrate when completely transitioning

1. **Any custom Impact content**
2. **Any custom WebGUI modifications or additions**
3. **Move users**
4. **Over time, probes get repointed.**

Remember, refer to the Best Practices document.

It’s really all the same stuff, hidden by a thin layer of clouds.