# BPPM 9.5 Enabling Custom KMs Best Practices

bmcsoftware

Participant passcode: 597271

Int'l Toll and US Cell Phone: 913-312-1392
US/CAN Toll free: 855-483-3538

## LOCAL:

Local - Australia, Brisbane: +61 (0) 7 3123 0047
Local - Australia, Canberra: +61 (0) 2 6111 2007
Local - Australia, Melbourne: +61 (0) 3 9034 3217
Local - Australia, Sydney: +61 (0) 2 9126 3023
Local - Austria, Vienna: +43 (0) 1 2675 904
Local - Bahrain, Manama: +973 1619 9040
Local - Belgium, Brussels: +32 (0) 2 303 2273
Local - Brazil, Sao Paolo: +55 11 3163 0461
Local - Bulgaria, Sofia: +359 (0) 2 491 6411
Local - Canada, Montreal: +1 514 669 6112
Local - Canada, Toronto: +1 647 426 9209
Local - China: +86 400 120 2693
Local - China: +86 400 120 2693
Local - Czech Republic, Prague: +420 234 147 003
Local - Denmark, Copenhagen: +45 78 78 79 63
Local - Estonia, Tallinn: +372 622 5742
Local - Finland, Helsinki: +358 (0) 9 7479 0101
Local - France, Lille: +33 (0) 359 69 03 42
Local - France, Lyon: +33 (0) 426 10 30 24
Local - France, Paris: +33 (0) 1 70 71 29 53
Local - Germany, Berlin: +49 (0) 30 2555 5430
Local - Germany, Frankfurt: +49 (0) 69 1200 9864
Local - Germany, Munich: +49 (0) 89 1436 7911
Local - Hong Kong, Hong Kong: +852 3008 0383
Local - Hungary, Budapest: +36 1 577 9957
Local - India, Bangalore: +91 (0) 80 6127 5134
Local - India, Mumbai: +91 (0) 22 6150 2334
Local - Ireland, Dublin: +353 (0) 1 437 0560
Local - Israel, Tel Aviv: +972 (0) 3 721 9373
Local - Italy, Milan: +39 02 9978 1800
Local - Italy, Rome: +39 06 8743 4377
Local - Japan, Tokyo: +81 (0) 3 4455 1996
Local - Latvia, Riga: +371 6601 3678
Local - Lithuania, Vilnius: +370 5205 5590
Local - Luxembourg, Luxembourg: +352 2786 0224
Local - Malaysia, Kuala Lumpur: +60 (0) 3 7724 0847
Local - Mexico, Mexico City: +52 55 4777 2663
Local - Netherlands, Amsterdam: +31 (0) 20 262 0137
Local - New Zealand, Auckland: +64 (0) 9 929 1884
Local - Norway, Oslo: +47 21 95 32 33
Local - Poland, Warsaw: +48 (0) 22 295 36 31
Local - Portugal, Lisbon: +351 21 120 9698
Local - Romania, Bucharest: +40 (0) 21 529 1340
Local - Russian Federation, Moscow: +7 495 620 9818
Local - Singapore, Singapore: +65 6416 9957
Local - Slovakia (Slovak Republic), Bratislava: +421 (0) 2 3278 6632
Local - Slovenia, Ljubljana: +386 (0) 1 888 8397
Local - South Africa, Johannesburg: +27 11 589 8382
Local - Spain, Madrid: +34 91 080 0153
Local - Sweden, Stockholm: +46 (0) 8 4030 4953
Local - Switzerland, Geneva: +41 (0) 22 555 0258
Local - Switzerland, Zurich: +41 (0) 44 556 8481
Local - Taiwan, Taipei: +886 (0) 2 2650 7292
Local - United Kingdom, London: +44 (0) 20 8150 0796

## TOLL FREE:

Int'l toll free - Argentina: 0800 666 2571
Int'l toll free - Australia: 1 800 635 764
Int'l toll free - Austria: 0800 295 994
Int'l toll free - Bahamas: 800 205 6295
Int'l toll free - Belgium: 0 800 72 785
Int'l toll free - Brazil: 0800 891 8445
Int'l toll free - Bulgaria: 00 800 115 1110
Int'l toll free - Chile: 123 0020 9601
Int'l toll free - China, Northern Region: 10 800 714 1853
Int'l toll free - China, Southern Region: 10 800 140 1378
Int'l toll free - Colombia: 01 800 518 0504
Int'l toll free - Czech Republic: 800 142 277
Int'l toll free - Denmark: 8088-6980
Int'l toll free - Dominican Republic: 1 888 751 4437
Int'l toll free - France: 0 800 914 669
Int'l toll free - Germany: 0 800 182 4414
Int'l toll free - Greece: 00 800 161 2205 1068
Int'l toll free - Hong Kong: 800 908 710
Int'l toll free - Hungary: 06 800 184 72
Int'l toll free - India: 000 800 100 7363
Int'l toll free - Indonesia: 001 803 011 2660
Int'l toll free - Ireland: 1 800 760 204
Int'l toll free - Israel: 1 80 945 2068
Int'l toll free - Italy: 800 873 593
Int'l toll free - Japan: 00531 12 0058
Int'l toll free - Korea, Republic Of: 00798 14 800 4574
Int'l toll free - Latvia: 800 03 656
Int'l toll free - Lithuania: 88 003 0200
Int'l toll free - Luxembourg: 800 25 164
Int'l toll free - Malaysia: 1 800 812 644
Int'l toll free - Mexico: 001 800 514 1058
Int'l toll free - Monaco: 800 93 304
Int'l toll free - Netherlands: 0 800 024 9645
Int'l toll free - New Zealand: 0 800 440 611
Int'l toll free - Norway: 800 191 83
Int'l toll free - Panama: 00 800 226 7179
Int'l toll free - Peru: 0800 55 444
Int'l toll free - Philippines: 1 800 111 00400
Int'l toll free - Poland: 00 800 112 41 37
Int'l toll free - Portugal: 800 819 382
Int'l toll free - Russian Federation: 810 800 2544 1012
Int'l toll free - Singapore, Singapore: 800 101 1738
Int'l toll free - Slovakia: 0800 606 314
Int'l toll free - Slovenia: 0 800 80815
Int'l toll free - South Africa: 0 800 999 562
Int'l toll free - South Korea, Korea, Republic Of: 003 0813 1654
Int'l toll free - Spain: 900 941 991
Int'l toll free - Sweden: 02 079 3671
Int'l toll free - Switzerland: 0 800 896 853
Int'l toll free - Taiwan: 00 801 126 826
Int'l toll free - Thailand: 001 800 12 066 3284
Int'l toll free - Trinidad and Tobago: 800 205 7179
Int'l toll free - United Arab Emirates: 800 017 7175
Int'l toll free - United Kingdom: 0 800 051 7166
Int'l toll free - Uruguay: 000413 598 2551
Int'l toll free - Venezuela: 0 800 100 2525

Participant passcode: 597271

NOTE: Some locations do not support cell phones and require a land line.

# BPPM 9.5 Best Practice
# Enabling Custom KMs

# Overview

- First Level Training - Basic Knowledge
- Best Practice & How To
- Covers Core BPPM Components
- Does not address every scenario
- Prior knowledge of BPPM components and terms

# Agenda

- Background & Assumptions
- Required Tools
- KM Objects Edited or Created
- Process Workflow
- Enabling Data Collection
- Enabling Policy Configuration
- Enabling Menu Commands
- General Recommendations

# Background & Assumptions

- Detailed Technical Knowledge
  - BPPM 9.5 Infrastructure & Architecture
  - PATROL 9.5 Agent Functionality & Configuration
  - BPPM 9.5 CMA Functionality
    - Policies
    - Repository
    - Servers & IS Nodes
  - BPPM Operational UI and navigation
  - PATROL Classic Console Developer Mode
  - PATROL Agent Configuration & pconfig DB
  - General PATROL KM Structure and Capabilities
  - Custom KM design and functionality
- PATROL KM Development Experience
  - Not optional
  - Previous experience is highly recommended

# Required Tools

- BPPM 9.5 Server with CMA installed and operational
- BPPM 9.5 Integration Service node installed and operational
- PATROL Classic Console version 3.6.00.1or higher
- BPPM 9.5 PATROL Agent installed and integrated with the BPPM 9.5 infrastructure
- PATROL Common Installer Generator Utility (PCIG)
- A common compression utility such as *zip* delivered with RHEL 6.4

# Required Tools

- Configure and setup all tools as you normally would.

- All of these tools should be installed in a development environment.

- Do not attempt to configure KMs for BPPM 9.5 in a production environment

- Do not install or use other tools in the development environment that may interfere with the development process.
  - Older PATROL Classic Console versions
  - Older BPPM infrastructure components
  - PATROL Central Consoles

# KM Objects Edited or Created

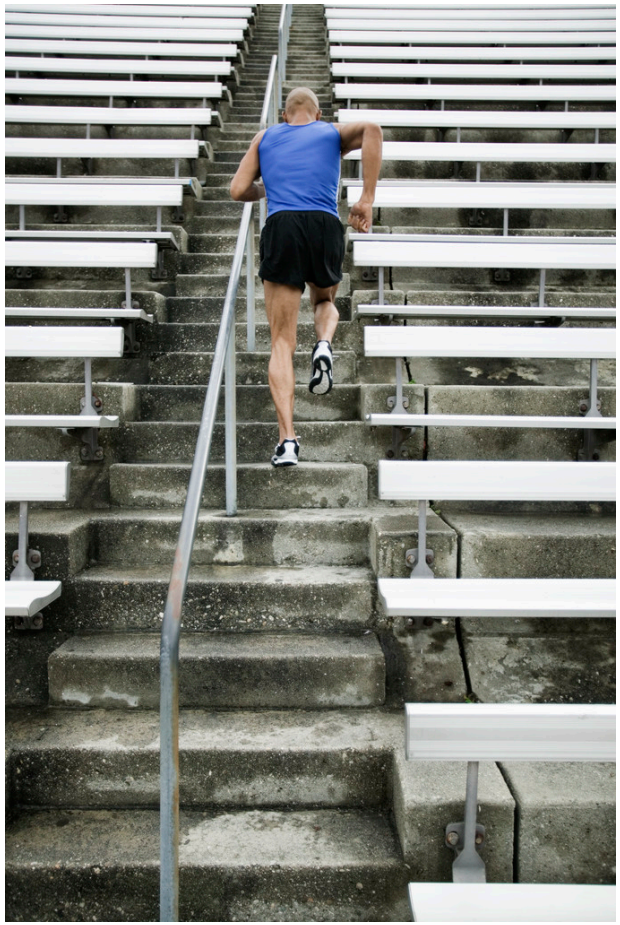
| Object Type | Description | Comments |
|---|---|---|
| KM file(s) | All the *.KM file(s) that make up a single custom Knowledge Module solution. | |
| KML file | A single *.KML file that lists all the Knowledge Module files that make up the custom Knowledge Module solution. | This is used to preload the Knowledge Module solution as with previous versions of PATROL. You should have one and only one KML file per Knowledge Module solution. |
| PSL file(s) | All *.psl files related to the Knowledge Module solution | These files may not be present. They are not required if they do not exist. If they exist they are required. |
| LIB file(s) | All *.lib files related to the Knowledge Module solution | These files may not be present. They are not required if they do not exist. If they exist they are required. |
| CFG file | *.cfg file containing agent configuration rules for the Knowledge Module solution. | Typically you will have a single CFG file for a single Knowledge Module solution. This file is not required but is recommended for any default settings. |
| XML files | *.XML files related to the Knowledge Module files | Each KM file will have a corresponding XML file. The XML files are generated by the PATROL Classic Console *knowledge* directory |
| Menu Command ID file(s) | *.lib file(s) related to menu commands | Each menu command that you want to enable in the BPPM UI requires a unique file. |
| bmc_products directory | Directory created by the PCIG utility that contains the monitoring solution install | |
| <KMNAME>_solution.zip file | Solution package created by the PCIG utility that you can import into the BPPM 9.5 CMA Monitoring Repository | Follow a standard naming convention |
| <KMNAME>.zip file | Silent install package | You create this package in the BPPM 9.5 CMA *Deployable Packages Repository*. |

# Process Workflow

- Enable custom KMs for BPPM 9.5 in three major phases in the order listed
  - Data Collection Enablement
  - Policy Configuration Management Enablement
  - Menu Command Enablement
- Test and validate results of each phase before moving on to the next phase
- Optional or Phased Enablement
  - If not enabling, Policy Configuration Management
    - Leverage rules in the Configuration Variables section of a monitoring policy
    - Or leverage PATROL & PCM
  - Menu Commands
    - Enable a subset
  - The KM can still collect data into the BPPM 9.5 server
  - General recommendation is to enable all that is necessary & possible

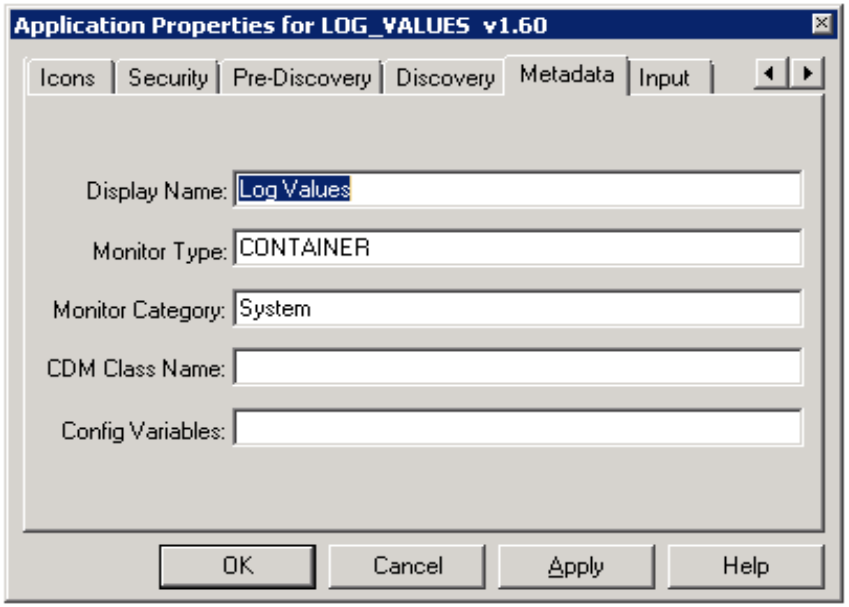
# Enabling Data Collection

- Validate the custom KM operates as designed and expected
  - Do this before any editing
  - This is irrespective of BPPM
- Launch the PATROL Classic Console in Developer mode and load the custom Knowledge Module.
  - Load any dependent Knowledge Modules as well
- Configure the agent to operate in non-Policy mode
- Ensure the agent is not connected to a BPPM Integration Service node
  - You will connect it later

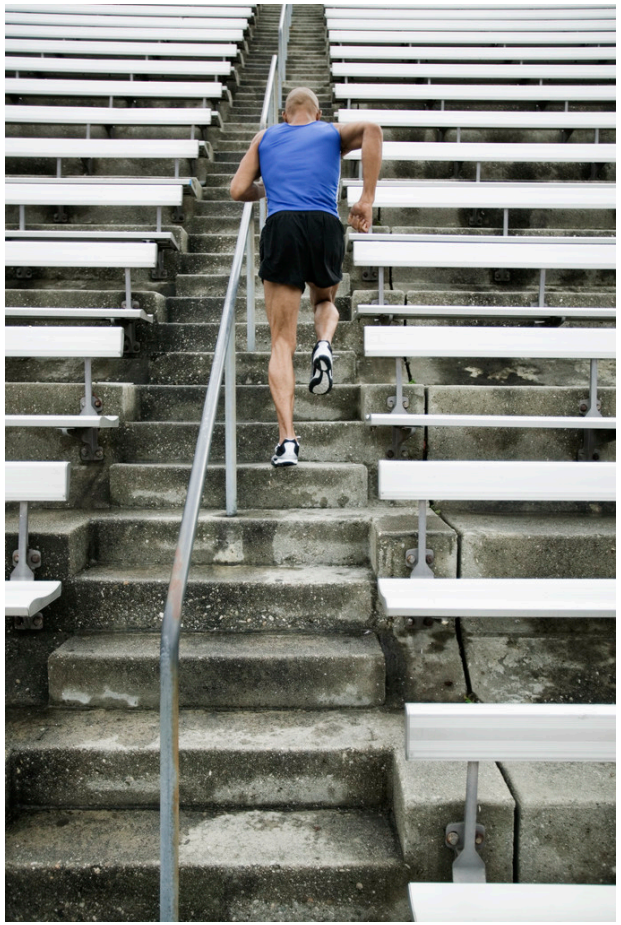


# Enabling Data Collection

- Add meta data settings to the KM properties.
  - For “Container” KMs specify CONTAINER
  - Sub KM application classes specify Monitor

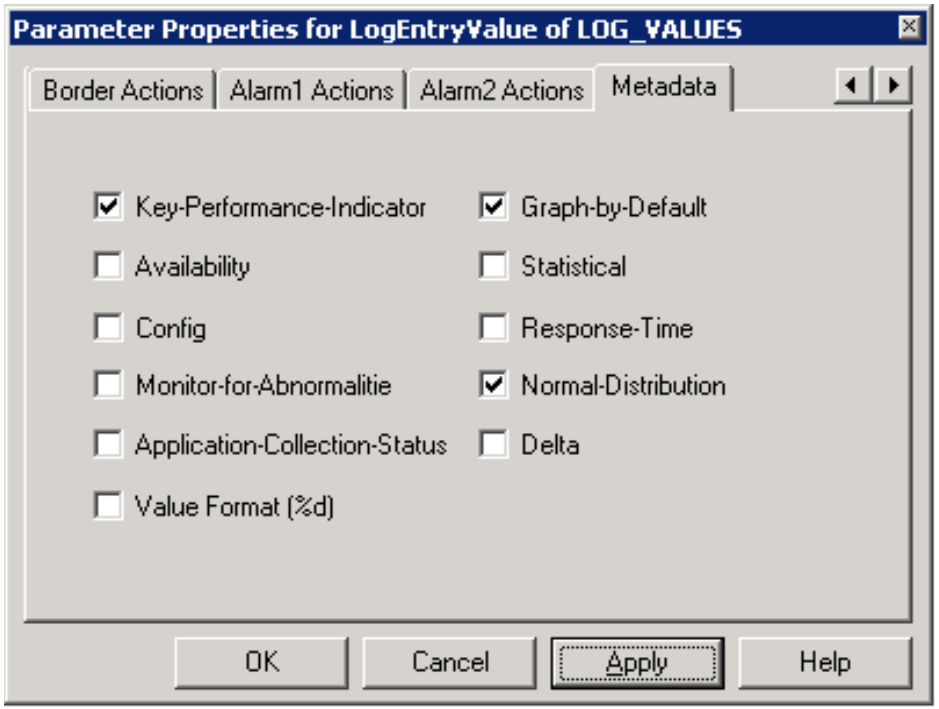


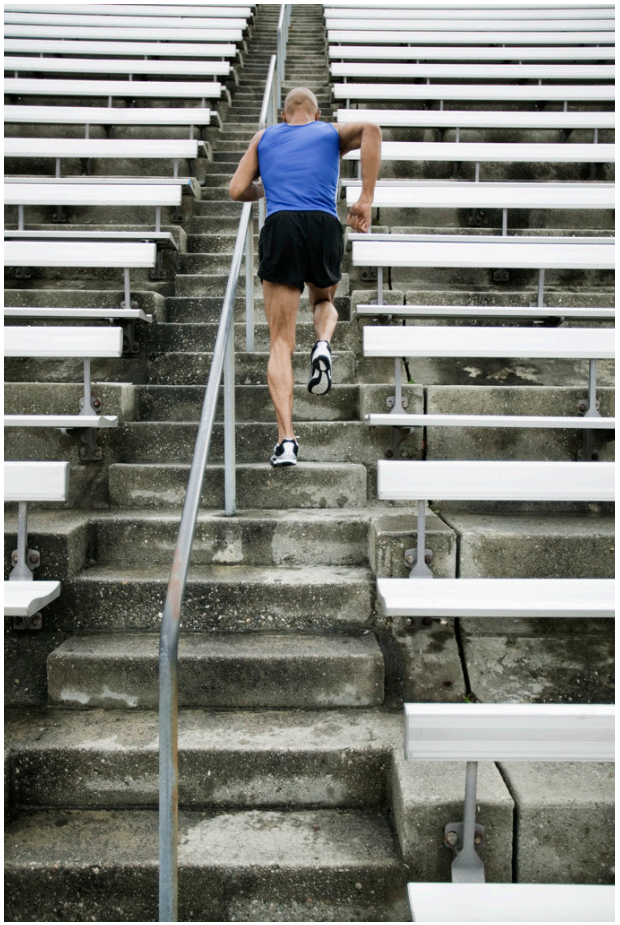

- Implement a container for each custom KM application class
  - Helps manage multiple instances
  - Will improve UI navigation in BPPM
  - Like instances are grouped into a container.



# Enabling Data Collection

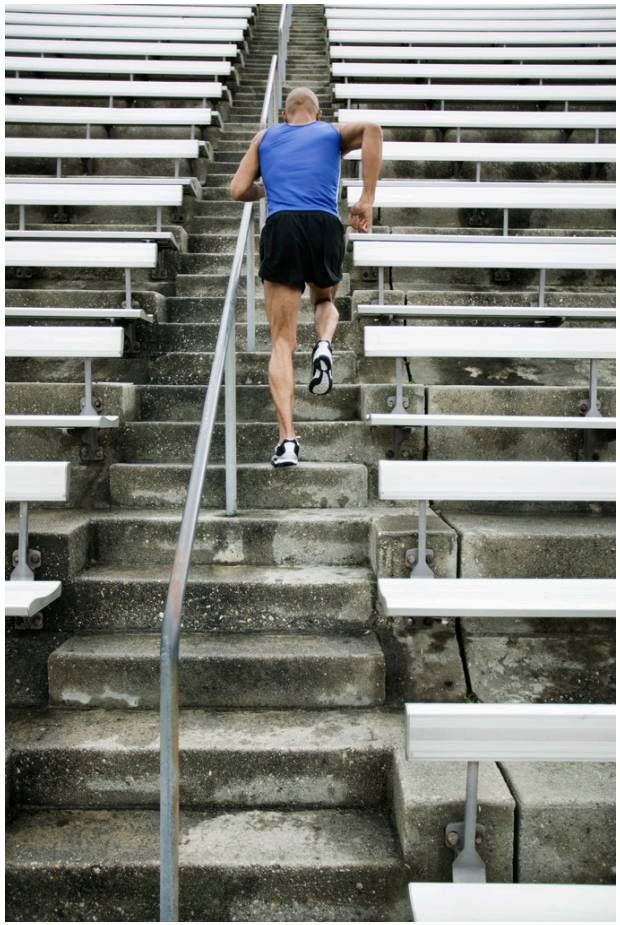
- Add meta data settings to the KM parameters.
  - For  Boolean metrics specify Availability
  - For performance metrics specify abnormality and/or other trend settings
  - Specify KPIs as appropriate
  - Etc.



# Enabling Data Collection

- Save PSL code in PSL files using the "Save to file" option in developer console
  - Instead of storing the code in the KM files
  - Reduces the size of the KM XML file
- Save the KM in the PATROL console
- Commit the KM
- Validate that the Knowledge Module continues to monitor properly in PATROL
- Exit the PATROL Classic Console
- Edit or create a KML file for the KM solution
  - Add all *.KM files to be loaded
  - Add an entry for MetaKMLDisplayName
  - Add an entry for MetaKMLDescription



# Enabling Data Collection

```
!PATROLV3.6.0.01i  FF08BBD43AC9210A8C6C41C85A315D3F
!++
!
! PATROL Session Knowledge Module
!
!--
! MetaKMLDisplayName  = "Log Values"
! MetaKMLDescription  = “KM that monitors numeric values from log file annotations”
KM_LIST = {
        "LOG_VALUES.km"
}
! 9999
```

# Enabling Data Collection

- Edit the Knowledge Module file (*.KM)

```
!PATROLV3.6.00.1i
3EC3A0E66408BF2FDB85D058CB4597BBE7EC56BBDB4BCA4A2D59D74EE5B65E7F
!#MSG_DOMAIN          km_sec
!++
!
! PATROL Session Knowledge Module
!
!--
!RELEASE          1.0.00
!REVISION         00
!PACKAGE          plv
!DESCRIPTION      PATROL Knowledge Module for Log Values
!PRODUCTCODE          plv

!PRAGMA allow: all
```

# Enabling Data Collection

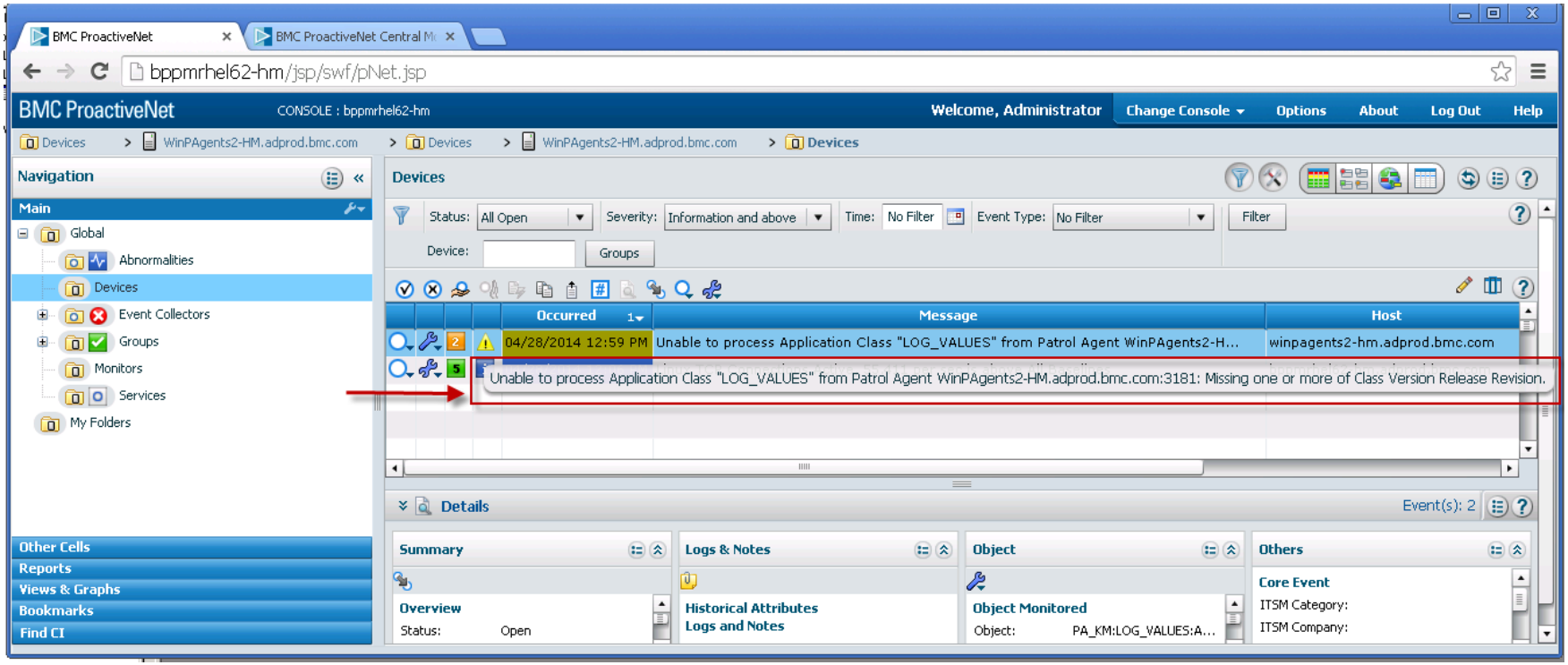
- Both the PACKAGE value and the PRODUCTCODE should be a unique value specific to the custom KM

- Do not use BMC Software package values and/or product codes.
  - Verification Source
    - BPPM CMA server
    - solution_list.xml file
    - Solution_Mapping.xml
    - Locations
      - UNIX/Linux:
        - $BPPM_SERVER_HOME/pw/pproxy/depot_directory/bmc_products
      - Windows:
        - %BPPM_SERVER_HOME%/pw/pproxy/depot_directory/bmc_products

- Include a letter combination that is unique to the company as part of the string
  - Example: “AIT” for Acme IT

# Enabling Data Collection

- If you do not enter the release and revision information into the KM file(s) the related application classes will not be processed in the BPPM server.
- When the agent connects to the IS node the application classes without release and revision information will be rejected and an error event will be generated in the BPPM UI.



# Enabling Data Collection

- Copy all the files related to the custom KM to a backup directory
  - KML file
  - KM files
  - PSL files
  - LIB files
  - Etc.
- Apply agent configuration rules to the agent
  - Custom KML in the KM preloaded list
  - Configure the agent to connect to the BPPM 9.5 test Integration Service node
- Stop the PATROL Agent
- Delete all the files related to the custom KM from…
  - PATROL Agent directories
  - PATROL Classic Console cache
  - You will be reinstalling them using a silent install package created in CMA

# Enabling Data Collection

- Create a subdirectory for the custom KM under the location where you have installed the PCIG utility.

```
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# pwd
/root/pcig/LINUX-X64-64
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# ls
cleanup.sh  jserver.start  LOGVALUES  out  pcig  rate.start
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
```

- Create file specific subdirectories for all the KM files under the location where you have installed the PCIG utility.

```
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# ls
cleanup.sh  jserver.start  LOGVALUES  out  pcig  rate.start
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# find LOGVALUES -type d
LOGVALUES
LOGVALUES/lib
LOGVALUES/lib/psl
LOGVALUES/lib/knowledge
LOGVALUES/LOGVALUES
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
```

# Enabling Data Collection

- Copy all the KM files from the backup location to their respective directories under the directories you created for the custom KM

```
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# pwd
/root/pcig/LINUX-X64-64
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# ls
cleanup.sh  jserver.start  LOGVALUES  out  pcig  rate.start
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# find LOGVALUES -type f
LOGVALUES/lib/psl/log_values_D.psl
LOGVALUES/lib/psl/log_values_PD.psl
LOGVALUES/lib/psl/log_values_MC.psl
LOGVALUES/lib/psl/log_values_Coll.psl
LOGVALUES/lib/knowledge/LOG_VALUES.kml
LOGVALUES/lib/knowledge/LOG_VALUES.xml
LOGVALUES/lib/knowledge/LOG_VALUES.km
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
```

# Enabling Data Collection

- Run the PCIG utility from the directory it is installed in

  *./pcig -p plv -r 1.0.00 -s LOGVALUES/ -l ALL -t LogValues*

| Argument | Purpose | Example | Comments |
|---|---|---|---|
| -p | product | plv | Enter a value that does not match or conflict with any other products. |
| -r | release number | 1.0.00 | Enter a 3 segment value. Do not enter only one or two segments. (For example 1 and 1.0 would not work.) |
| -s | location of all source files and their respective sub directories | *LOGVALUES /* | Use this syntax exactly with the proper subdirectory name followed by a forward slash at the end |
| -l | comma separated list of supported OS types | ALL | Make sure this matches the KM design |
| -t | product title name | LogValues | This name is used and displayed in the BPPM CMA repository UI as the name of the solution in the *Monitoring Repository*. It is also the name of the compressed ZIP file that is generated. |

# Enabling Data Collection

- The PCIG utility will create a *bmc_products* directory and a ZIP file for the solution

```
[root@BPPMRHEL62-HM LINUX-X64-64]# ./pcig -p plv -r 1.0.00 -s LOGVALUES/ -l ALL -t LogValues
VCI0102: Preparing PPF [HEADER] section...
VCI0101: Loading files for packaging...
        7 files prepared for packaging.
VCI0103: Preparing PPF [FILES] section...
VCI0104: Generating PPF file... bmc_products/Index/plvskm.ppf...
VCI0105: Generating CAT file... bmc_products/Products/plvskm/plvskm.cat...
VCI0106: Compressing...
VCI0107: Image is ready.
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# ls
bmc_products  jserver.start  LOGVALUES  out  pcig  plv.zip  rate.start
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
```

NOTE: In the current PCIG pre-release version the zip file will not work. Compress the *bmc_products* directory using a ZIP utility instead.

# Enabling Data Collection

```
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# zip -r LogValues.zip bmc_products
  adding: bmc_products/ (stored 0%)
  adding: bmc_products/Products/ (stored 0%)
  adding: bmc_products/Products/unixfam/ (stored 0%)
  adding: bmc_products/Products/unixfam/unixfam.xml (deflated 65%)
  adding: bmc_products/Products/unixfam/unixfam.ppf (deflated 48%)
  adding: bmc_products/Products/unixfam/unixfam.cat (stored 0%)
  adding: bmc_products/Products/plvskm/ (stored 0%)
  adding: bmc_products/Products/plvskm/plvs5.gz (stored 0%)
  adding: bmc_products/Products/plvskm/plvs6.gz (stored 0%)
  adding: bmc_products/Products/plvskm/plvs7.gz (stored 0%)
  adding: bmc_products/Products/plvskm/plvs2.gz (stored 0%)
  adding: bmc_products/Products/plvskm/plvs3.gz (stored 0%)
  adding: bmc_products/Products/plvskm/plvs4.gz (stored 0%)
  adding: bmc_products/Products/plvskm/plvs1.gz (stored 0%)
  adding: bmc_products/Products/plvskm/plvskm.cat (deflated 59%)
  adding: bmc_products/Products/plvskm/plvskm.ppf (deflated 48%)
  adding: bmc_products/Products/mswfam/ (stored 0%)
  adding: bmc_products/Products/mswfam/mswfam.xml (deflated 66%)
  adding: bmc_products/Products/mswfam/mswfam.cat (stored 0%)
  adding: bmc_products/Products/mswfam/mswfam.ppf (deflated 51%)
  adding: bmc_products/Index/ (stored 0%)
  adding: bmc_products/Index/mswfam.ppf (deflated 51%)
  adding: bmc_products/Index/plvskm.ppf (deflated 48%)
  adding: bmc_products/Index/unixfam.ppf (deflated 48%)
  adding: bmc_products/imginfo.000000-000000 (deflated 10%)
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]# ls
bmc_products  jserver.start  LOGVALUES  LogValues.zip  out  pcig  plv.zip  rate.start
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
[root@BPPMRHEL62-HM LINUX-X64-64]#
```

# Enabling Data Collection

- Use different names for the solution package created by the PCIG utility and the silent install package you create in the BPPM 9.5 CMA Deployable Package Repository.

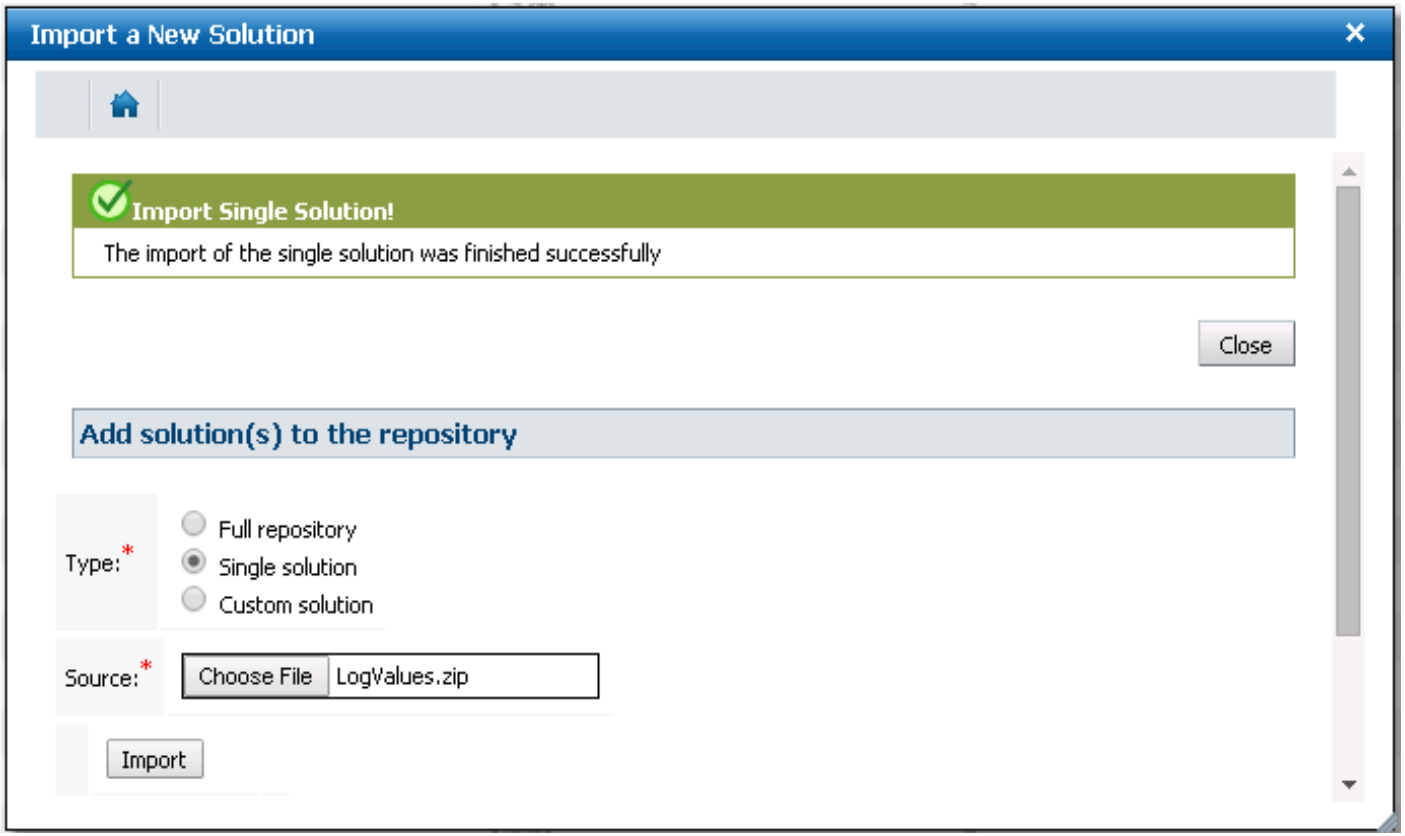
  - Name the file created by the PCIG utility based on the product name
    - Example: plv.zip – “plv” is short for Patrol Log Value
  - Name the deployable package according to the KM name
    - Example: LogValues.zip

- Do not use product names that match or conflict with other product names

- Do not use package or solution names that match or conflict with other package or solution names

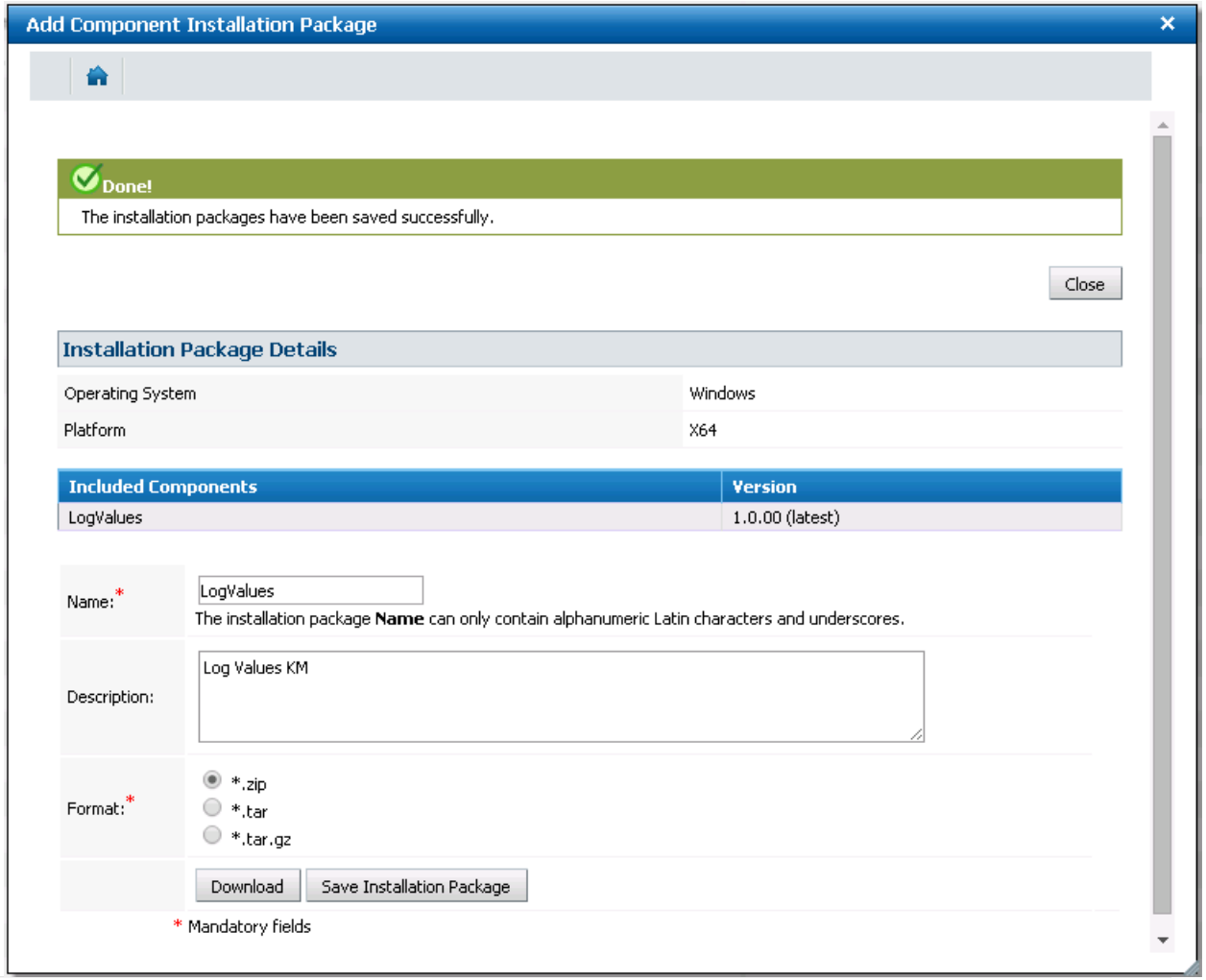
# Enabling Data Collection

- Import the Zip file you created from PCIG into the BPPM 9.5 CMA Monitoring Repository as a single solution.



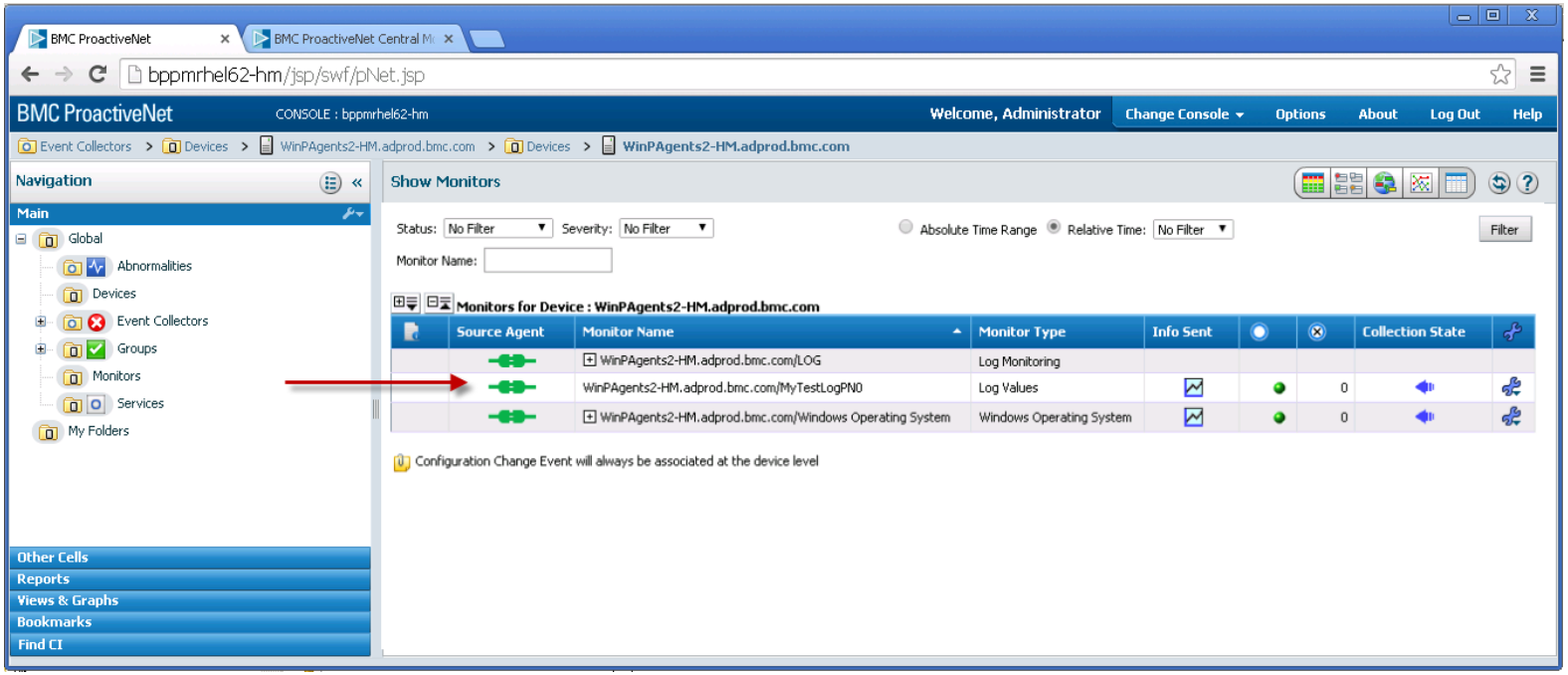
# Enabling Data Collection

- Create a silent install package in the BPPM 9.5 CMA Deployable Package Repository
  - Unique Name
  - Meaningful Name
  - Complete Description
  - Do not include the agent
  - Do not include other KMs
  - Follow Configuration Best Practices



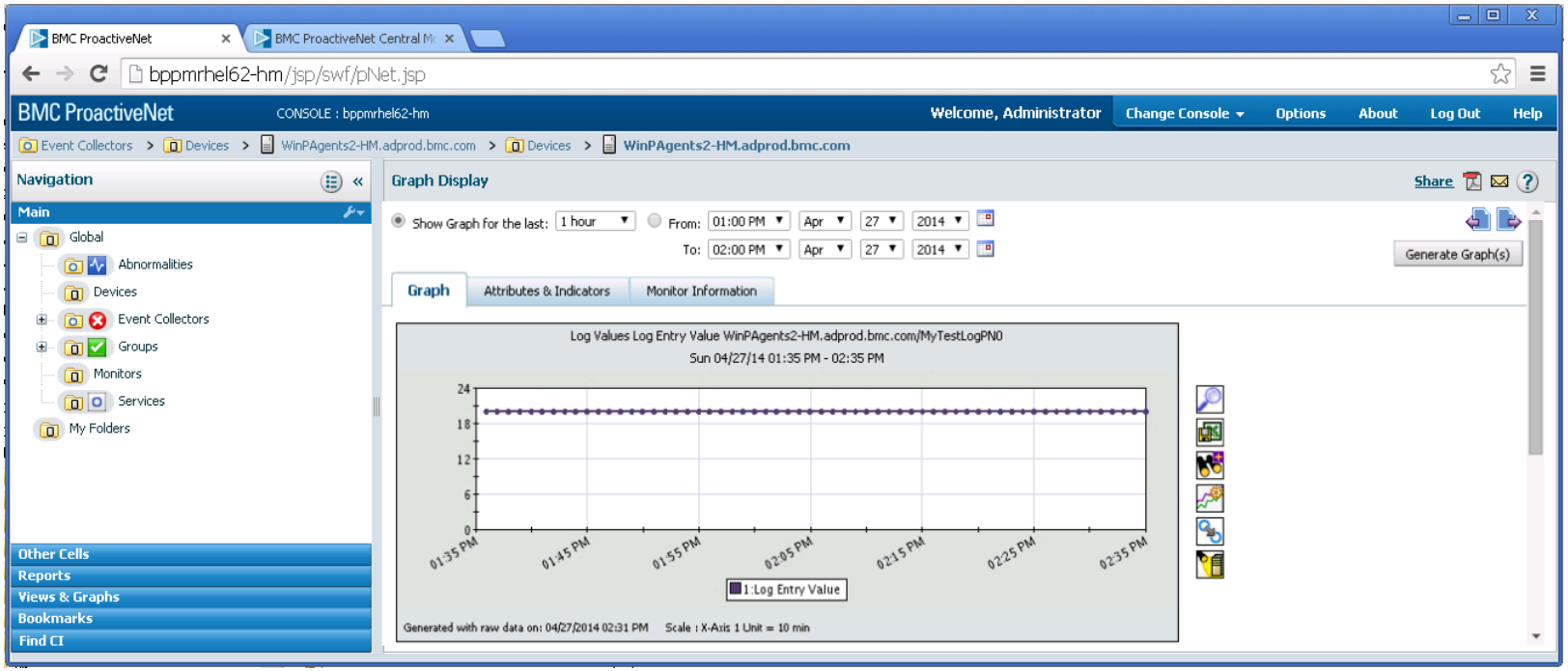
# Enabling Data Collection

- Download the silent install package and move it to the test managed node
- Run the silent installer for the package
- Start the PATROL Agent
- Verify that the KM application class(es) appear(s) as monitor type(s) in the BPPM server operations UI



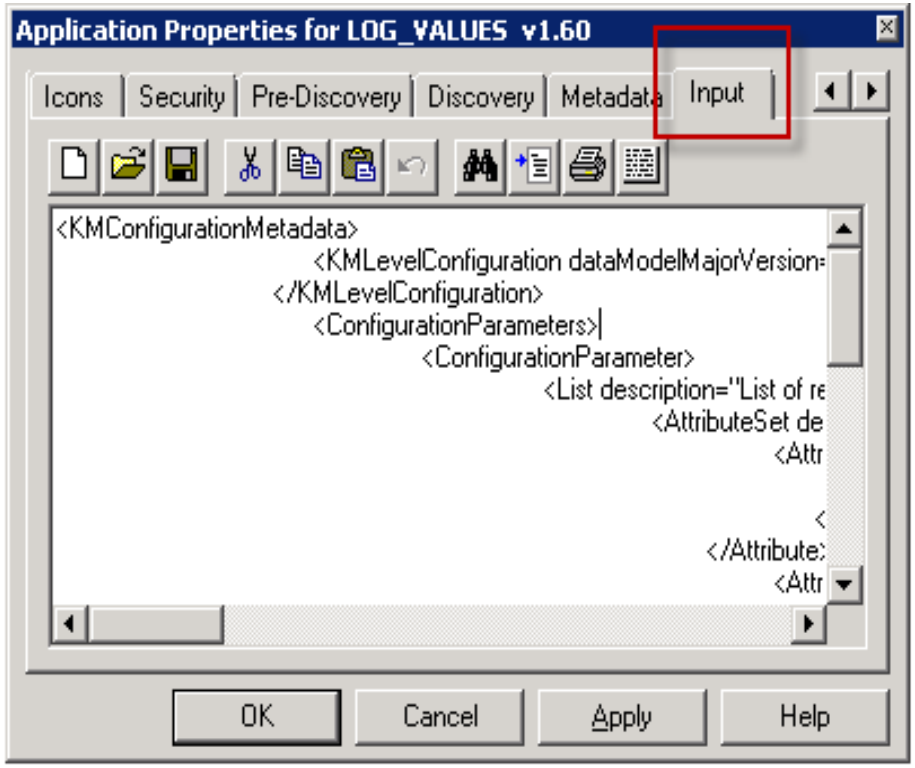
# Enabling Data Collection

- Validate data collection before moving on to enabling policy configuration
- Consider all aspects, for example annotation, etc.



# Enabling Policy Configuration

- Requirements
  - Define attributes for configuration properties
  - Create XML data for the attributes
  - Include Host level constructs in the XML data
  - Enter the XML data into the KM application class properties on the Input tab



# Enabling Policy Configuration

- Much of the structure and content of the XML data is the same for various KMs
  - Host configuration related section
  - Various labels.
  - You can easily copy existing Input XML data and edit it

- XML data unique to a custom
  - Titles
  - Content s in the attribute sets
  - Attributes that define the specific configuration for the custom KM.

- The XML Input data is saved into the KM XML file in the Console Cache *knowledge* directory

# Enabling Policy Configuration

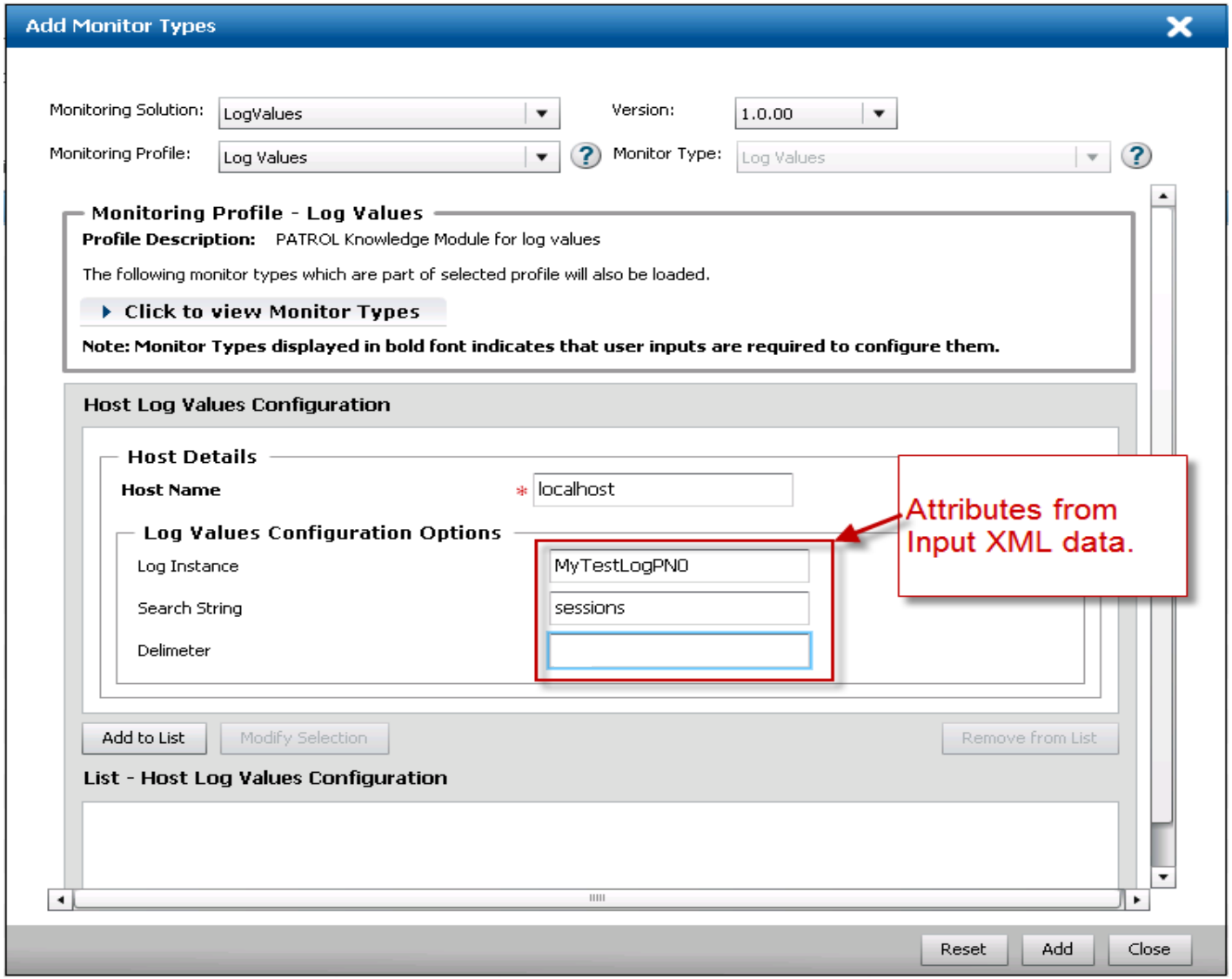
```
<KMConfigurationMetadata>
<KMLevelConfiguration dataModelMajorVersion="1" dataModelMinorVersion="1">
</KMLevelConfiguration>
<ConfigurationParameters>
<ConfigurationParameter>
<List description="List of remote hosts" id="HOSTS" indexedBy="host" isMandatory="true" label="Host Log Values Configuration">
<AttributeSet description="Enter Host Name for which the specified configuration applies" i18nIdForDescription="4" i18nIdForLabel="3"
      id="HostConfiguration" label="Host Details">

<Attribute>
<String default="localhost" description="Enter the Host Name" i18nIdForDescription="6" i18nIdForLabel="5" id="host" isMandatory="true"
      label="Host Name">
</String>
</Attribute>
<Attribute>
<AttributeSet id="LogValues" label="Log Values Configuration Options">
        <Attribute>
                <String default="" description="Enter the log instance to monitoring." id="LVInstance" isMandatory="false" label="Log Instance">
                </String>
        </Attribute>
        <Attribute>
                <String default="" description="Enter the alarm search string value." id="LVSearchString" isMandatory="false" label="Search String">
                </String>
        </Attribute>
        <Attribute>
                <String default="" description="Enter the data delimiter." id="LVDelimiter" isMandatory="false" label="Delimeter">
                </String>
        </Attribute>
</AttributeSet>
</Attribute>
</AttributeSet>
</List>
</ConfigurationParameter>
</ConfigurationParameters>
</KMConfigurationMetadata>
```

# Enabling Policy Configuration

- Practice and leverage examples to help you learn what data is needed
- Enter unique ID values for each attribute
- Enter concise labels
  - Use: “Search String”
  - Do not use: “Search String for Log Annoatation
- Enter concise titles
- Be cognizant of screen space
  - Keep labels as short as possible
  - Keep titles as short as possible
  - Leverage descriptions in the attributes for explanations (mouse over)
- Deployment
  - After entering XML data save the KM
  - Work through the standard packaging and import process using PCIG
  - The KM XML file goes in the *knowledge* directory for PCIG

# Enabling Policy Configuration



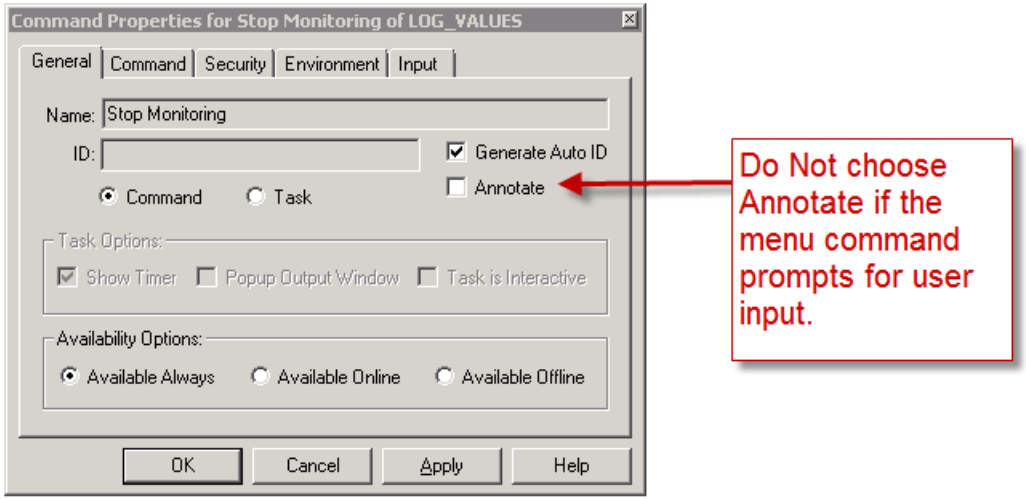
# Enabling Policy Configuration

```
PATROL_CONFIG
"/ConfigData/LOG_VALUES/HOSTS/localhost/host" = { REPLACE = "localhost" },
"/ConfigData/LOG_VALUES/HOSTS/localhost/LVDemiliter" = { REPLACE = " " },
"/ConfigData/LOG_VALUES/HOSTS/localhost/LVInstance" = { REPLACE = "MyTestLogPN0" },
"/ConfigData/LOG_VALUES/HOSTS/localhost/LVSearchString" = { REPLACE = "sessions" }
```

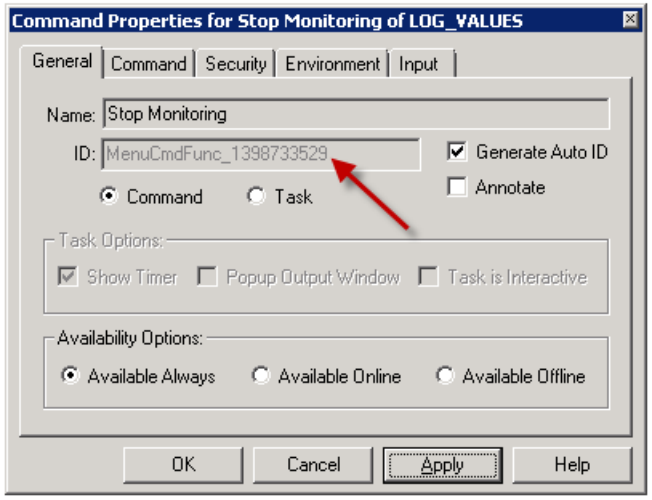
- KMs need to be edited where the pconfig PSL command is used
  - Get pconfig values
  - Write pconfig values
  - Consider structure of the /ConfigData/<KM name> branch
- Configure the agent to collect data into the BPPM server in non-policy mode first
- Review the results of policy configuration in pconfig and edit accordingly
- Test and validate policy configuration before moving on to enabling Menu Commands.

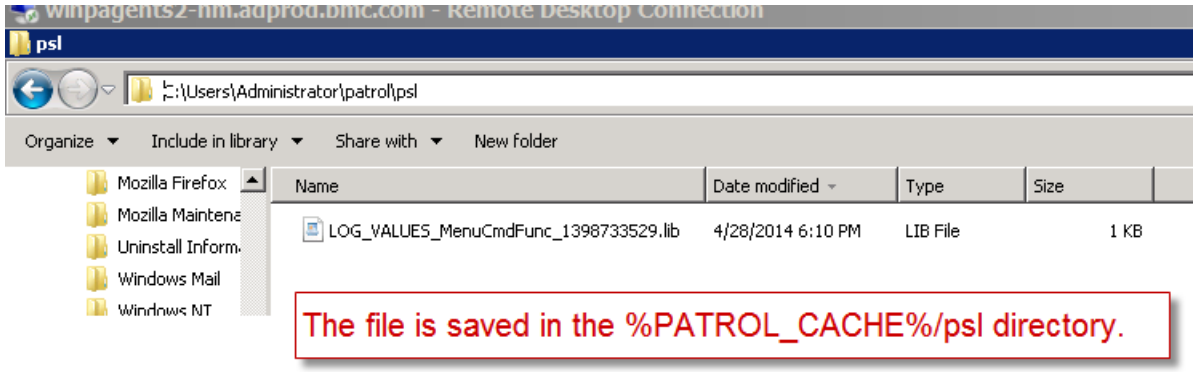
# Enabling Menu Commands

- Requirements
  - Copy and backup the Command code first
  - Edit properties for the Menu Command
    - Generate an ID for Menu Command
      - Creates a unique ID for the Command
      - Creates a *.lib file for the Command in the PSL directory of the Console Cache
        - » Each Menu Command will have a unique ID and *.lib file.
  - Include the *.lib file in the package for PCIG in the *psl* directory



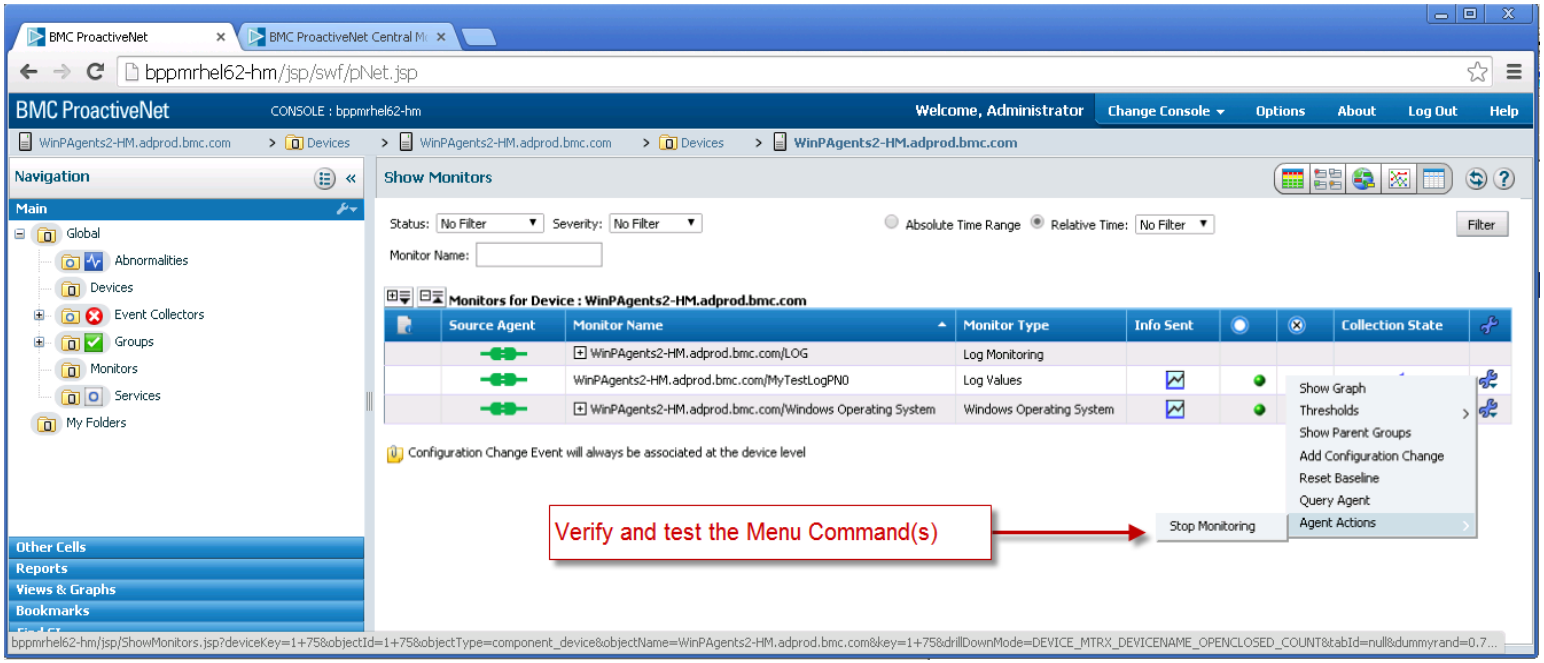
# Enabling Menu Commands



# Enabling Menu Commands

- Deployment
  - After generating the command ID save the KM
  - Work through the standard packaging and import process using PCIG
  - The *.lib file for the ID(s) goes in the *psl* directory for PCIG
- Validate that the Menu Command appears in the BPPM UI and functions properly



# Enabling Menu Commands

- Enable only Menu commands that you desire in the BPPM UI
  - Not all Menu Commands have to be enabled
  - If desired, you can enable no Menu Commands
  - Don't create unnecessary work

- Leverage the auto ID and file generation capability

- Select the Annotate option only if the Menu Command is to be automated

- Do not edit the ID file name

- Do not open and/or edit the ID file contents

# General Recommendations

- Make a backup of all KM related files before you begin editing.
- Start with a simple KM. Learn the process before moving on to more complicated KMs.

- Keep your code organized and follow the recommended process.
- Do not attempt to combine multiple KM solutions into one package.
- Do not jump ahead. Do not skip steps.

- Do not work though the enablement process in production.
- Do not use KM names that match or conflict with other KM names including BMC product names.

# General Recommendations

- Observe and follow general Knowledge Module development best practices.

- Observe and follow general software development best practices.
- Establish, document, and follow a release, version and revision numbering methodology.
- Do not manually edit files except where recommended and required.

- Do not use *Microsoft WordPad* or *Microsoft Word* for editing/saving files.
  - Use *vi* on Linux and UNIX
  - Use *Notepad* or a programmer's text editing utility like *Notepad*++ on Windows

# General Recommendations

- Work in stages and take backups often before moving on to the next stage

- Maintain release and revision numbers properly
  - Important if you plan to have multiple release and/or revision numbers in use over time
  - Also helps track a development process

- Start with a working Knowledge Module in PATROL

- Do not try to mix Knowledge Module development for monitoring and development/editing for BPPM 9.5 enablement

- Do not use the Microsoft Windows compressed folders to compress KM packages

# Additional Resources & Information

- Online Documentation
  - BPPM 9.5 Best Practices
    https://communities.bmc.com/docs/DOC-28658
  - BPPM 9.5 Product Documentation
    https://docs.bmc.com/docs/display/public/proactivenet95/Home

- BMC Communities (public forum)
  - BMC website
    - documents
    - discussions
    - whitepapers
    - additional information
  - https://communities.bmc.com/communities/community/bmcdn/service_assurance
  - Recording of today's presentation and content
    - https://communities.bmc.com/docs/DOC-28658
    - Follow/Subscribe to the page to be notified about updates


bmcsoftware
BUSINESS RUNS ON I.T.
I.T. RUNS ON BMC™