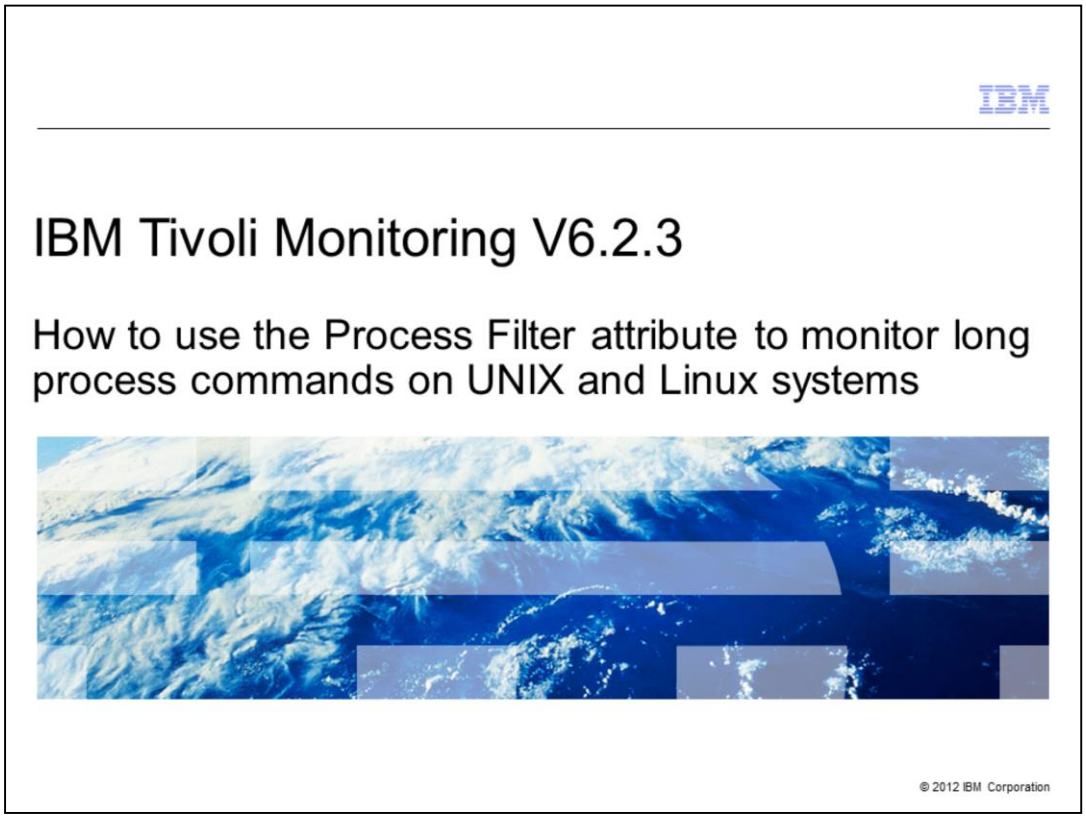
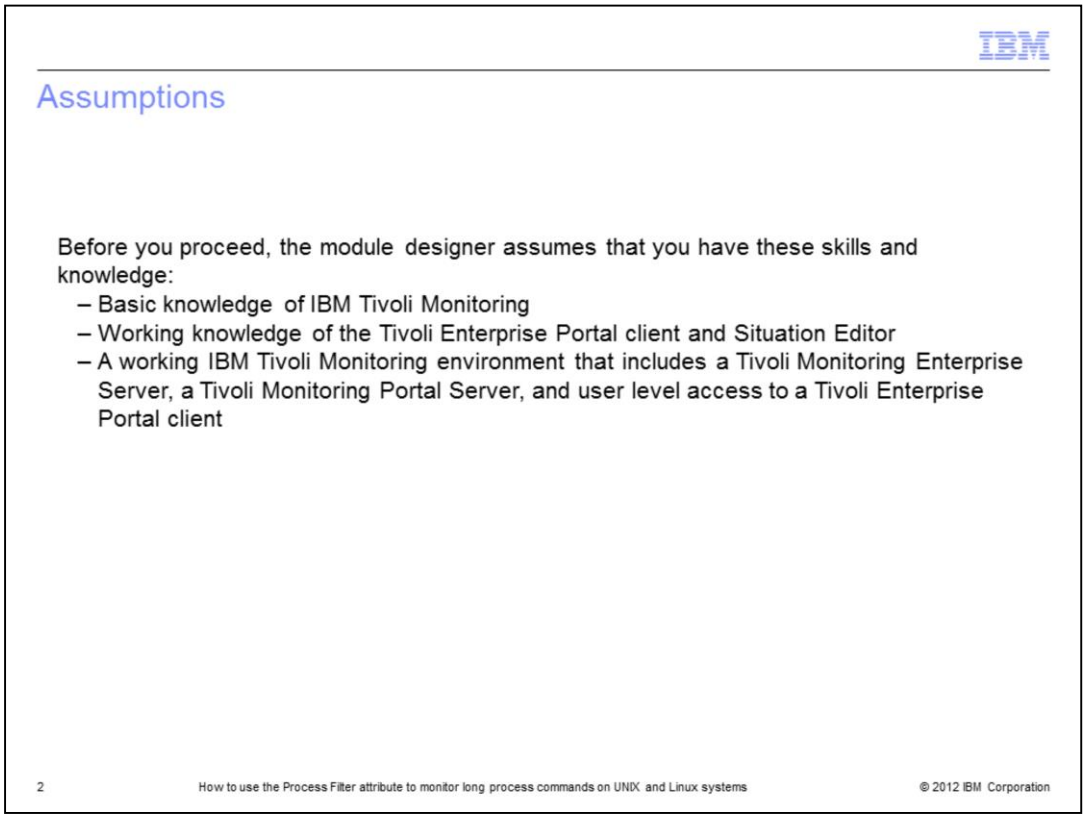


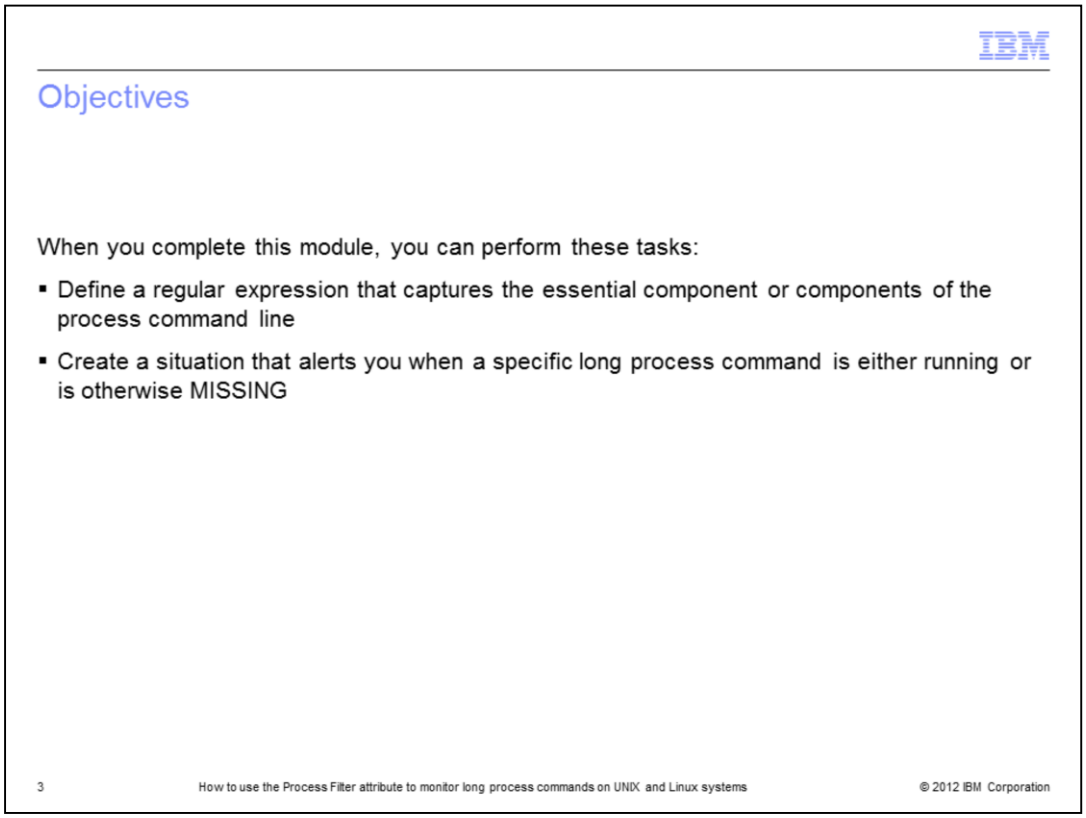
IBM Tivoli® Monitoring V6.2.3, How to use the Process Filter attribute to monitor long process commands on UNIX® and Linux® systems. In this module, you learn how to create a situation that monitors long process commands.



The module developer assumes that you have these skills and knowledge:

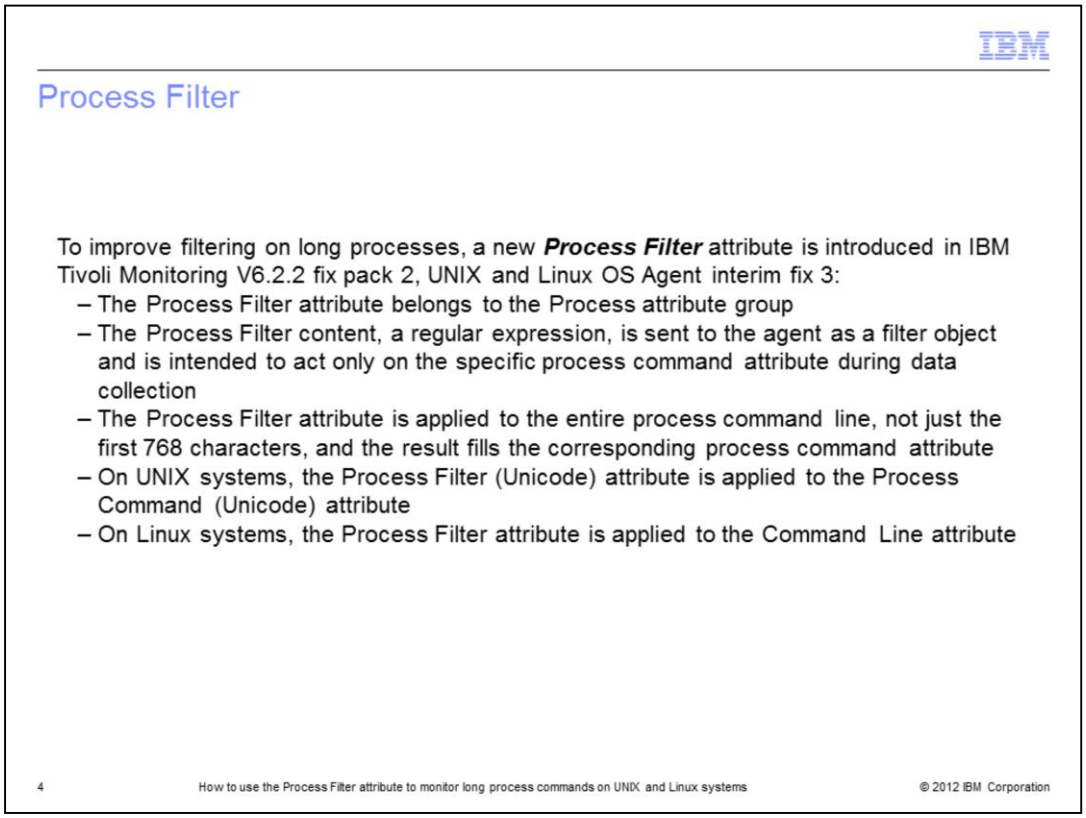
- Basic operating knowledge of IBM Tivoli Monitoring on UNIX and Linux operating systems

- A functioning IBM Tivoli Monitoring environment with at least user level access to the Tivoli Enterprise Portal client



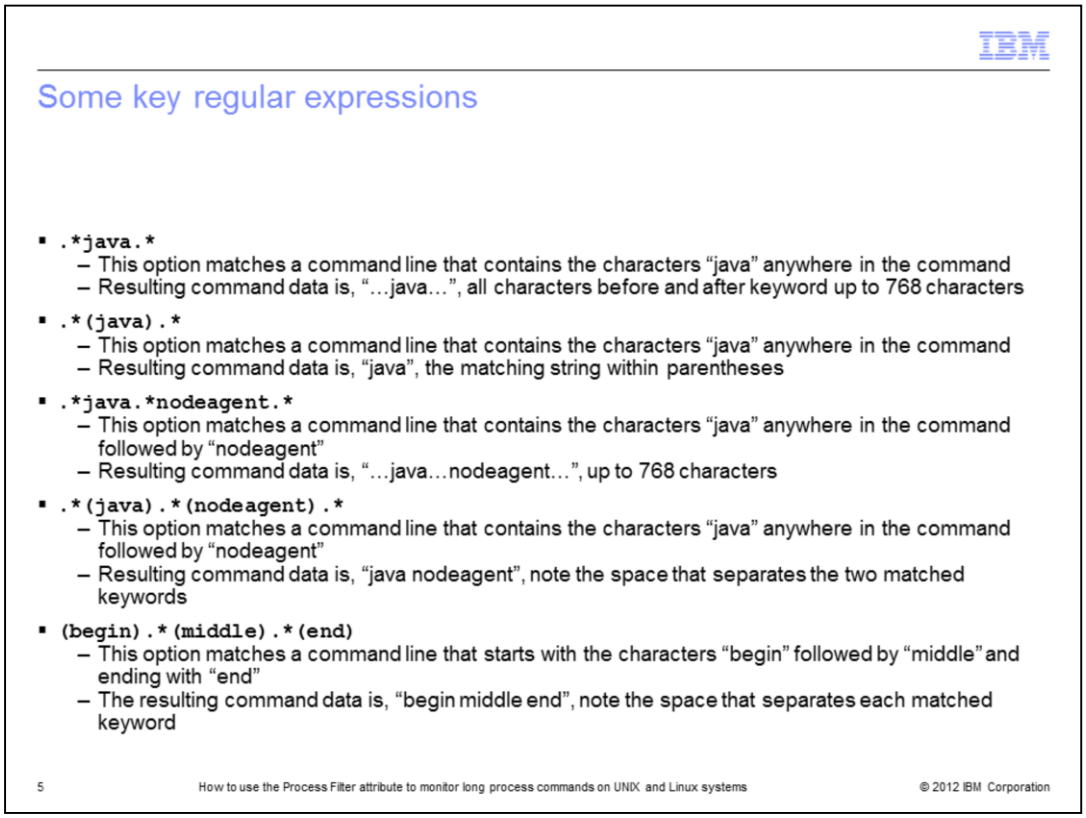
When you complete this module, you can perform these tasks:

- Define a regular expression that uniquely identify the required process command
- Create a situation that generates an alert when the process command is either running or is otherwise MISSING



To improve filtering on long processes, a new Process Filter attribute was introduced in IBM Tivoli Monitoring V6.2.2, fix pack 2, UNIX and Linux OS Agent interim fix 3. The Process Filter attribute belongs to the Process attribute group.

The Process Filter content, a regular expression, is sent to the agent as a filter object and is intended to only act on a specific process command attribute during data collection. Note the specific attribute that the Process Filter attribute is applied to on UNIX and Linux systems.



Here are several very useful regular expressions that you can use when attempting to match the required process command line. Notice carefully how each will produce different results in the corresponding command data based on the matching keyword with and without parentheses. The use of parentheses helps reduce the command line to just the keywords of interest such that it can be easily searched.

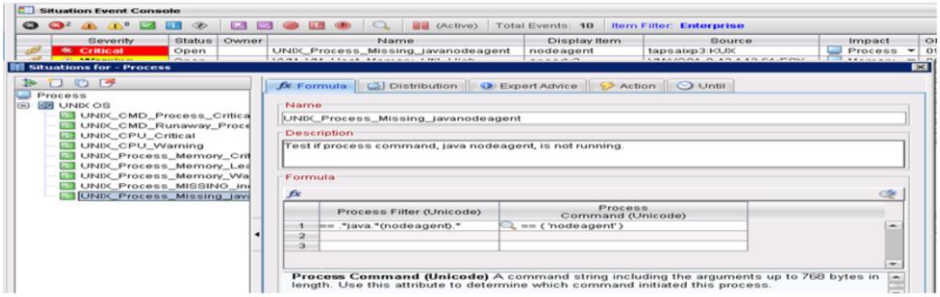
Process Command Monitor (UNIX OS Agent)

- A long Java process command:

```
/u02/IBM/WebSphere/AppServer/java/bin/java -Xbootclasspath/p:…
…policy=/u02/IBM/WebSphere/AppServer/profiles/profileManaged/properties/server.policy com.ibm.ws.bootstrap.WSLauncher
com.ibm.ws.runtime.WsServer
/u02/IBM/WebSphere/AppServer/profiles/profileManaged/configDU01
DU01dkencaap001a nodeagent
```

- Situation definition to alert when this process is not running or is otherwise MISSING:

(Process Filter (Unicode) == '.*java.*(nodeagent).*' AND MISSING(Process Command (Unicode)) == ('nodeagent'))



Here is an example of a process command running on a UNIX system. A situation defines a filter that uniquely identify the keywords in the process command. The use of the MISSING operator detects when the process is not running. The Situation Editor in the foreground show the actual definition. The situation event console in the background show the actual event alert.

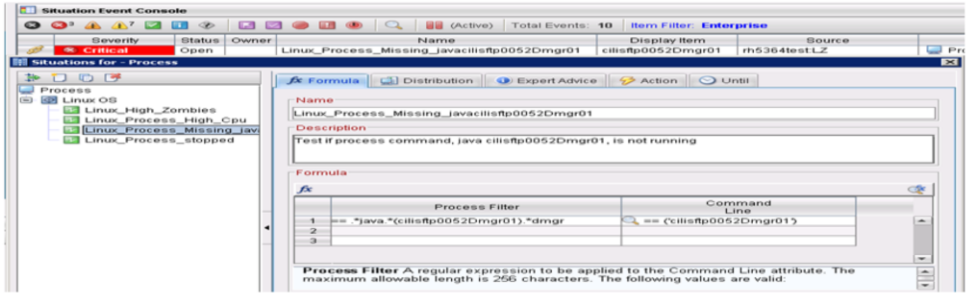
Process Command Monitor (Linux OS Agent)

- A long Java process command:

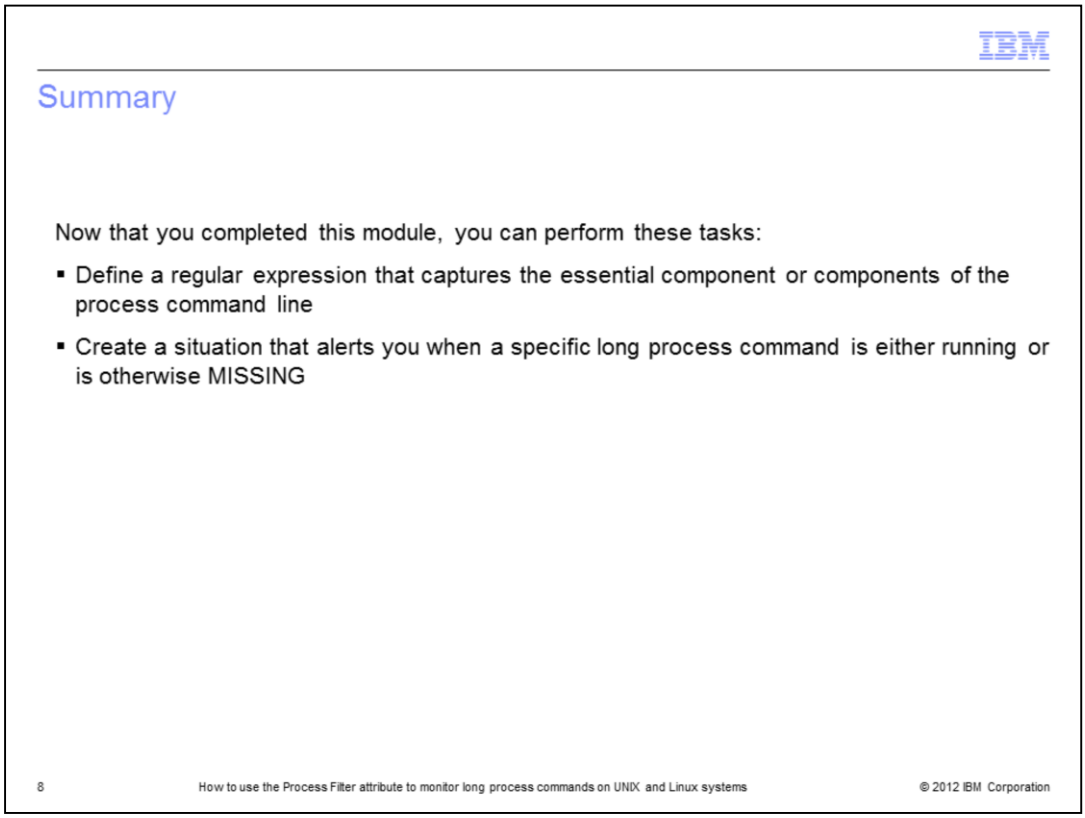
```
/opt/was70/AppServer/java/bin/java -Declipse.security -Dwas.status…
…policy=/opt/was70/AppServer/profiles/Dmgr01/properties/server.polic
y com.ibm.wsspi.bootstrap.WSPreLauncher -nosplash -application
com.ibm.ws.bootstrap.WSLauncher com.ibm.ws.runtime.WsServer
/opt/was70/AppServer/profiles/Dmgr01/config cilisftp0052Cell01
cilisftp0052Dmgr01 dmgr
```

- Situation definition to alert when this process is not running or is otherwise MISSING:

(Process Filter == .*java.*(cilisftp0052Dmgr01).*dmgr AND MISSING(Command Line) == ('cilisftp0052Dmgr01'))



Here is an example of a process command line running on a Linux system. A situation defines a filter that uniquely identify the keywords in the process command line. The use of the MISSING operator detects when the process is not running. The Situation Editor in the foreground show the actual definition. The situation event console in the background show the actual event alert.



Now you can perform the tasks to define a regular expression that uniquely identify the desired process command and create a situation that generates an alert when the process command is either running or is otherwise MISSING.

Trademarks, disclaimer, and copyright information