Creating Content Packs in vCenter Log Insight

TECHNICAL WHITE PAPER

Table of Contents

Introduction

Content packs are immutable, or read-only, plugins to vCenter™ Log Insight™ that provide pre-defined knowledge about specific types of events such as log messages. The goal of a content pack is to provide knowledge about a specific set of events in a format easily understandable by administrators, engineers, monitoring teams, and executives. A content pack should answer questions like, *"Is the product/application healthy?"* In addition, a content pack should create a greater understanding of how the product/application works.

A content pack is made up of information that can be saved from either the Dashboards or Interactive Analytics pages in Log Insight. This includes:

- Queries
- Fields
- Aggregations
- Alerts
- Dashboards

By default, the current version of Log Insight ships with the vSphere content pack, and other content packs can be imported as needed. In addition, any Log Insight user can create a content pack for private or public consumption.

Intended Audience

This paper provides information about each piece of information that can be saved in a content pack as well as best practices for content pack creation. The information provided is specifically tailored to content pack authors using Log Insight 1.0.

Getting Started

Before explaining how to create a content pack, it is important to understand some concepts regarding the content pack workflow. The tips in this section will make creating and maintaining content packs easier.

Instance

Content packs are immutable, or read-only, plugins to Log Insight, which means imported content packs cannot be edited. The easiest way to edit a content pack is to modify the saved definitions on the instance of Log Insight used to initially create the content pack. The original instance should be backed up to prevent data loss or corruption. If the instance used to create the content pack is lost and no backup exists, the content pack must be recreated on a new instance. While certain components of a content pack can be cloned into a custom dashboard, also known as user space, doing so is not recommended to edit a content pack and may result in a content pack that is dependent on a separate content pack.

User

Content packs are created in part from the content saved under Custom Dashboards, also known as user space, or more specifically either My Dashboards or Shared Dashboards on the Dashboards page and when exporting a content pack, everything within the selected custom dashboard is exported. For this reason, it is recommended that every individual content pack be authored by a separate user entity in Log Insight. For information on creating users in Log Insight, please refer to the Log Insight in-product documentation.

Events

It is essential to collect relevant events before attempting to create a content pack to ensure that a content pack covers all relevant events for a product/application. One common way to collect relevant events is to ask quality assurance (QA) and/or support teams as these teams usually have access to, and knowledge about, common events. Attempting to generate events while creating a content pack will be time consuming and will likely result in missing important events. If QA and support teams are unable to supply events, simulated events may be used instead if product/application events are known and/or documented.

Once appropriate logs have been collected, they must be ingested into Log Insight. While not supported in the current version of Log Insight, it is possible to ingest events from the command line using the same process as the archive import process described in the Log Insight *Installation and Administration Guide*. In short, any file, directory, tarball, or .ZIP can be ingested by copying the events to the Log Insight virtual appliance and running: */usr/lib/loginsight/application/bin/loginsight repository import /path/to/events*. While this process is not supported, it will work and is recommended when creating a content pack.

Authors

The authors of a content pack should have qualifications as outlined below.

- Experience using VMware vCenter Log Insight.
- Real world operating knowledge of the product/application.
- Understanding and ability to generate optimized regular expressions.
- Experience debugging multiple problems with product/application using logs.
- Support background, with exposure to a myriad of problems.
- System administrator background with previous syslog experience.

Queries

Log Insight allows for queries to retrieve and summarize events. Queries can be created and saved from the Interactive Analytics page. A query is made up of one or more of the following:

- **Keywords:** Complete, or full-text, alphanumeric and/or hyphen matches.
- **Globs:** Asterisk and/or question mark symbol used to match some quantity of keywords.
- **Regular expressions:** Sophisticated string pattern matching based on Java regular expressions.
- **Field operations:** Keyword, regular expression, and pattern matches applied to extracted fields.
- **Aggregations:** Functions that are applied to one or more subgroups of the results.

Log Insight supports the following types of queries:

- **Message:** A query made up of keywords, regular expressions and/or field operations.
- **Regular expression or field:** A query made up of keywords and/or regular expressions.
- **Aggregation:** A query made up of a function, one or more groupings, and any number of fields.

Custom alerts can be defined in Log Insight and are triggered from scheduled queries of any type.

Saving Queries

Queries can be saved via one or more of the following:

- **Add to Dashboard:** Saves the last run query without time range as a chart widget in a dashboard group on the Dashboards page.
- **Save Current Query:** Saves the last run query with time explicit time range as a loadable query on the Interactive Analytics page. Queries saved using Save Current Query and exported as part of a content pack do not include any time range.

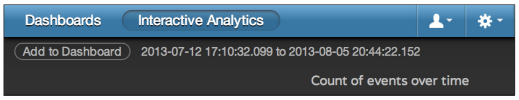

Dashboards
Interactive Analytics
Add to Dashboard
2013-07-12 17:10:32.099 to 2013-08-05 20:44:22.152
Count of events over time

Figure 1. Note the **Add to Dashboard** link just below the navigation bar on the Interactive Analytics page.

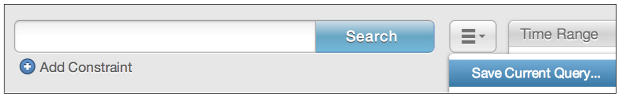
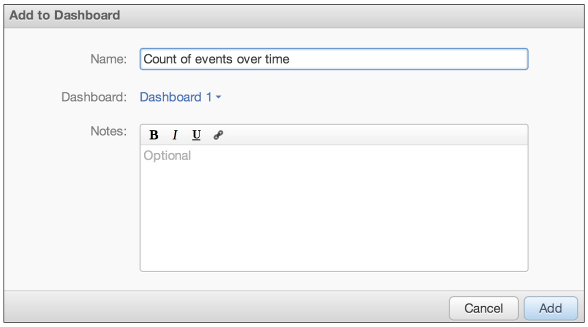


Figure 2. The **Save Current Query…** link under the menu drop-down on the Interactive Analytics page.

The notes section is very important and should be populated for every query. Information added can be text, a link to documentation, a knowledge base article, or a forum. Information provided should answer the following questions:

- Why is this widget important?
- What is a "good" and "bad" value?
- Where can more information be obtained?



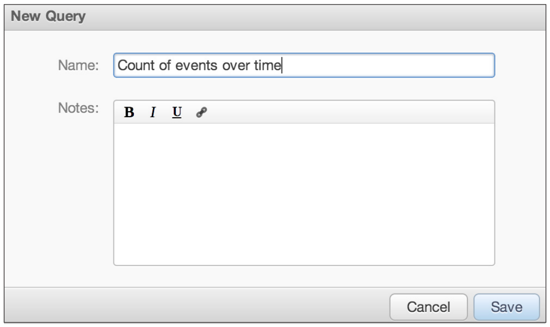
Add to Dashboard

Name: Count of events over time

Dashboard: Dashboard 1

Notes: Optional

Figure 3. Add to Dashboard dialog box with notes section.



New Query

Name: Count of events over time

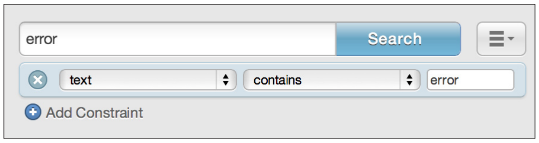
Notes:

Figure 4. Save Current Query dialog box with notes section.

Message Queries

Message queries can be entered via one or more of the following:

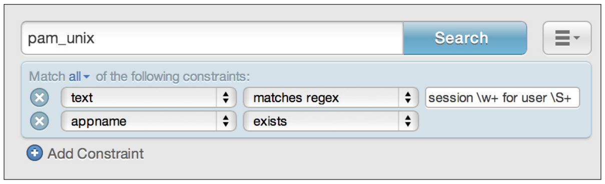
- **Search bar:** The search bar is one way to refine the results returned given the existing events in a Log Insight instance. While a constraint can be used instead of the search bar, it is often easier to understand a query that leverages the search bar over an equivalent constraint. As such, the best practice is to use the search bar versus an equivalent constraint whenever possible.
- **Constraints:** A constraint allows for querying using a regular expression, a field, logical OR operation, or a combination of search bar and constraint queries.



error

text | contains | error

Figure 5. An example of the search bar with a keyword and a constraint with an equivalent query. The search bar is preferred.



pam_unix

Match all of the following constraints:

text | matches regex | session \w+ for user \S+

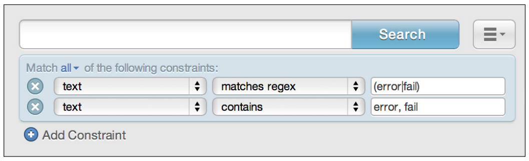
appname | exists

Figure 6. An example of the search bar with a keyword, a constraint with a regular expression, and a constraint with a field operation. In order for the query to return, all three items would need to return a match.

While query building is outside the scope of this document, there are several important things to know about the search bar and constraints when creating content packs. In general, the following best practices apply:

- **When constructing a query, use keywords whenever possible, when keywords are not sufficient use globs, and when globs are not sufficient use regular expressions.** Keyword queries are the least resource intensive type of query. Globs are a simplified version of regular expression and are the next least resource intensive type of query. Regular expressions are the most intensive type of query.
- **Avoid regular expressions whenever possible.** If the query can be written without regular expressions then it should be written without regular expressions primarily because, from a resource perspective, regular expressions are the most intensive type of query. Leverage globs instead of regular expressions when keywords are not sufficient.

- **Provide as many keywords as possible.** When using regular expressions or fields, be sure to include as many keywords as possible. Any keywords should be outside of any regular expressions including a logical OR like: (*this*|*that*). Regular expressions use a lot of resources. Keyword queries are the least resource intensive type of query and Log Insight is optimized to perform keyword queries prior to regular expressions to minimize regular expression overhead.

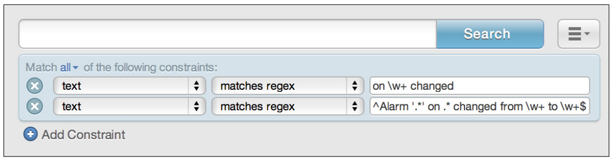


Match all of the following constraints:

text | matches regex | (error|fail)

text | contains | error, fail

Figure 7. An example of two different ways to construct the same query. The first constraint is a regular expression while the second is a keyword, comma separated, logical OR match. The second constraint is always preferred over the first.



Match all of the following constraints:

text | matches regex | on \w+ changed

text | matches regex | ^Alarm '.*' on .* changed from \w+ to \w+$

Figure 8. An example of two different ways to query for the same field. The first constraint is generic and only contains two keywords. The second constraint is very specific and has five keywords. The second constraint is always preferred over the first.

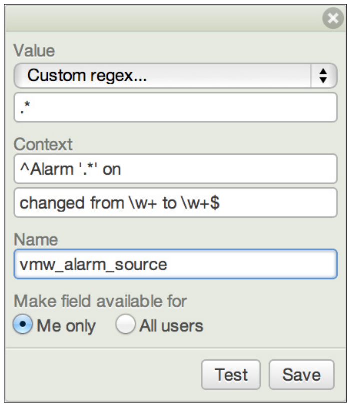
Field Queries

Fields are a powerful way to add structure to unstructured events and allow for the manipulation of both the textual and visual representation of data. Fields are one of the most important items in a content pack as they can be used in multiple ways including:

- **Aggregations:** Allow for functions and groupings to be applied to fields.
- **Constraints:** Allow for operations to be performed against fields.

Any part of a log message that might be applicable to a query or aggregation should be extracted. Fields are a type of regular expression query and are especially useful for complex pattern matching so the user does not need to know, remember, or learn complicated regular expressions.

- **Regex before value:** This field should include as many keywords as possible. If this field is empty or only contains special characters, then the *Regex after value* must include keywords.
- **Regex after value:** This field should include as many keywords as possible. If this field is empty or only contains special characters, then the *Regex before value* must include keywords.
- **Name:** Only use alphanumeric characters. Ensure all characters are lower case and use underscores instead of spaces as this makes fields easier to view. Important: Names for content pack fields and user fields can be the same though content pack fields will have a namespace in parenthesis to the right of the field name. It is recommended to prefix content pack fields with an abbreviation (e.g. vmw_) to avoid confusion.



Value

Custom regex...

.*

Context

^Alarm '.*' on

changed from \w+ to \w+$

Name

vmw_alarm_source

Make field available for

● Me only ○ All users

Figure 9. An example of an extracted field definition with multiple keywords.

In addition to the various components that make up a field, several best practices must be considered. These include:

- **Only create fields for regular expression patterns.** If a field can be queried using keyword queries, then keyword queries should be used instead of a pre-defined field. Fields are meant to add structure to unstructured data as well as provide a way to query over specific parts of an event.

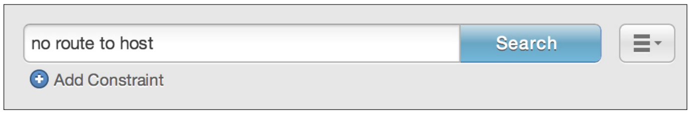

Search: no route to host

Figure 10. The recommended way to query for keyword matches. Information entered into the search bar or a constraint can also be saved for future usage by selecting the menu drop-down next to the Search button and selecting Save Current Query…

- **Only create fields for regular expression patterns that return a fraction of the total events.** Fields that will match most events and/or return a very large number of results are not a good candidate for field extraction because the regular expression will need to be applied to a large quantity of events resulting in a resource-intensive operation.

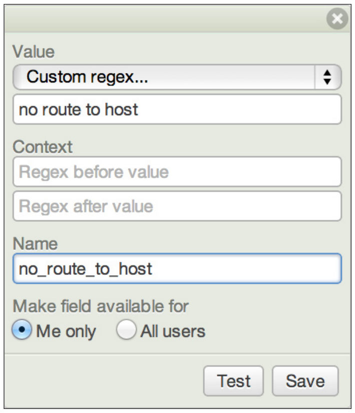
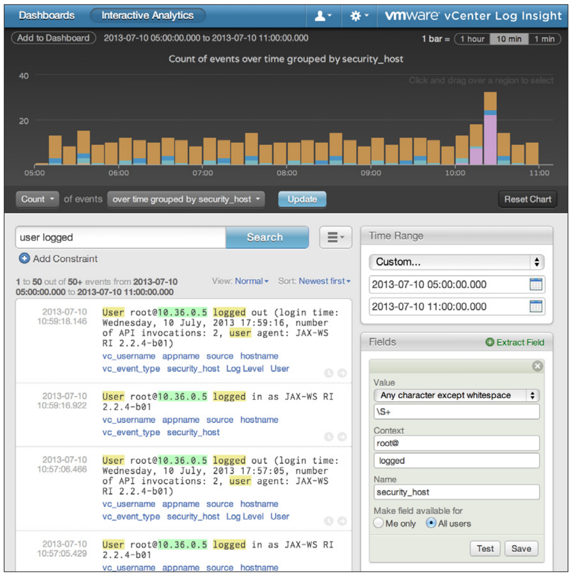

Value: Custom regex…
no route to host
Context: Regex before value / Regex after value
Name: no_route_to_host
Make field available for: ● Me only ○ All users

Figure 11. An example of a keyword field. Since this query can be constructed without a regular expression, it is not a good candidate for field extraction.

Orphaned Fields

It is common for queries to contain one or more fields. For saved queries, it is important to note that the field definition used when a query is saved is always maintained. This means if a query is saved with a field and that field is later modified, then the query will not be modified. Field modifications would include:

- The *value* of the field is changed
- The *regex before value* and/or the *regex after value* of the field is changed
- The *name* of the field is changed
- The field is deleted


Dashboards
Interactive Analytics
vmware vCenter Log Insight
Add to Dashboard
2013-07-10 05:00:00.000 to 2013-07-10 11:00:00.000
1 bar = 1 hour 10 min 1 min
Count of events over time grouped by security_host
40
20
Click and drag over a region to select
05:00
06:00
07:00
08:00
09:00
10:00
11:00
Count
of events
over time grouped by security_host
Update
Reset Chart
Search
Add Constraint
Time Range
Fields
Extract Field
Test
Save

Search: user logged

1 to 50 out of 50+ events from 2013-07-10 05:00:00.000 to 2013-07-10 11:00:00.000 — View: Normal — Sort: Newest first

Time Range: Custom… — 2013-07-10 05:00:00.000 — 2013-07-10 11:00:00.000

Value: Any character except whitespace — \S+
Context: root@ — logged
Name: security_host
Make field available for: ○ Me only ● All users

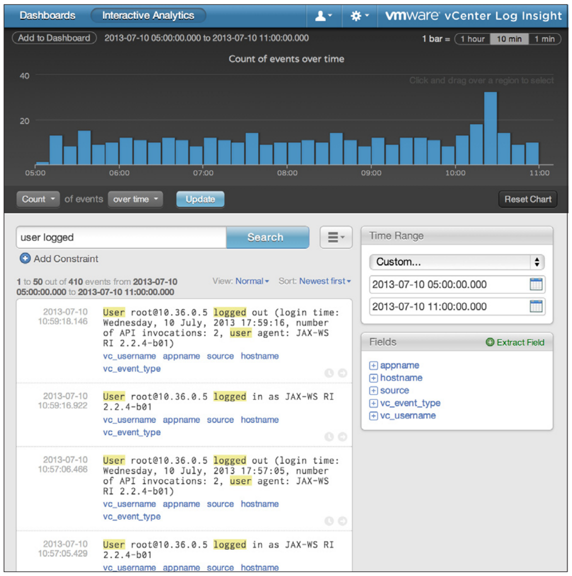
```
2013-07-10     User root@10.36.0.5 logged out (login time:
10:59:18.146   Wednesday, 10 July, 2013 17:59:16, number
               of API invocations: 2, user agent: JAX-WS
               RI 2.2.4-b01)
               vc_username appname source hostname
               vc_event_type security_host Log Level User

2013-07-10     User root@10.36.0.5 logged in as JAX-WS RI
10:59:16.922   2.2.4-b01
               vc_username appname source hostname
               vc_event_type security_host

2013-07-10     User root@10.36.0.5 logged out (login time:
10:57:06.466   Wednesday, 10 July, 2013 17:57:05, number
               of API invocations: 2, user agent: JAX-WS
               RI 2.2.4-b01)
               vc_username appname source hostname
               vc_event_type security_host Log Level User

2013-07-10     User root@10.36.0.5 logged in as JAX-WS RI
10:57:05.429   2.2.4-b01
               vc_username appname source hostname
```

Figure 12. An example of running a query with an orphaned field. Notice how the overview chart is grouped by security_host and the security_host field definition is open under the Fields section. This means the field does not exist in the Log Insight instance, but does exist as part of a chart widget or saved query.



Search: user logged

1 to 50 out of 410 events from 2013-07-10 05:00:00.000 to 2013-07-10 11:00:00.000 — View: Normal — Sort: Newest first

Time Range: Custom… — 2013-07-10 05:00:00.000 — 2013-07-10 11:00:00.000

Fields: appname, hostname, source, vc_event_type, vc_username

```
2013-07-10     User root@10.36.0.5 logged out (login time:
10:59:18.146   Wednesday, 10 July, 2013 17:59:16, number
               of API invocations: 2, user agent: JAX-WS
               RI 2.2.4-b01)
               vc_username appname source hostname
               vc_event_type

2013-07-10     User root@10.36.0.5 logged in as JAX-WS RI
10:59:16.922   2.2.4-b01
               vc_username appname source hostname
               vc_event_type

2013-07-10     User root@10.36.0.5 logged out (login time:
10:57:06.466   Wednesday, 10 July, 2013 17:57:05, number
               of API invocations: 2, user agent: JAX-WS
               RI 2.2.4-b01)
               vc_username appname source hostname
               vc_event_type

2013-07-10     User root@10.36.0.5 logged in as JAX-WS RI
10:57:05.429   2.2.4-b01
               vc_username appname source hostname
```

Figure 13. An example of what happens when an orphaned dialog box is closed. Notice how the search bar remains the same, but the overview chart is no longer grouped by security_host. In addition, security_host is not listed under the Fields section.

It is critical that saved queries that leverage a field be recreated if the field is modified. If a previously saved query is not updated when a field it relies on is modified, then the saved query will contain an orphaned field. An orphaned field is a field that exists in a saved query, but does not exist as an available field. Orphaned fields are apparent when running a saved query in the Interactive Analytics page, as a green field dialog box will be open under the *Fields* section. Important: Saving, deleting, closing, or modifying the open dialog box will result in any use of the orphaned field being removed from the query.

Ensure that content pack queries do not contain orphaned fields. If an orphaned field is found, recreate the saved query and delete the old saved query to remove the orphaned field. To remove an orphaned field from a chart widget:

- Go to the widget on the Dashboards page.
- Select the *Open in Interactive Analytics* arrow button within the widget.
- Modify the field(s) used.
- Select the *Add to Dashboard* button on the Interactive Analytics page.
- Select the *Delete* button from the gear button of the old widget from the Dashboards page.

Aggregation Queries

Log Insight allows for visual manipulation of events through the use of aggregation queries. An aggregation query is made up of two district attributes:

- Functions
- Groupings

In content packs, groupings are the most important consideration, but both functions and groupings will be addressed as they impact how charts are displayed. An aggregation query requires one function and at least one grouping.

Bar Charts

By default, the Interactive Analytics page of Log Insight displays a count of events over time in the overview chart. If the count function is used in conjunction with the time series grouping, then a bar chart is created.

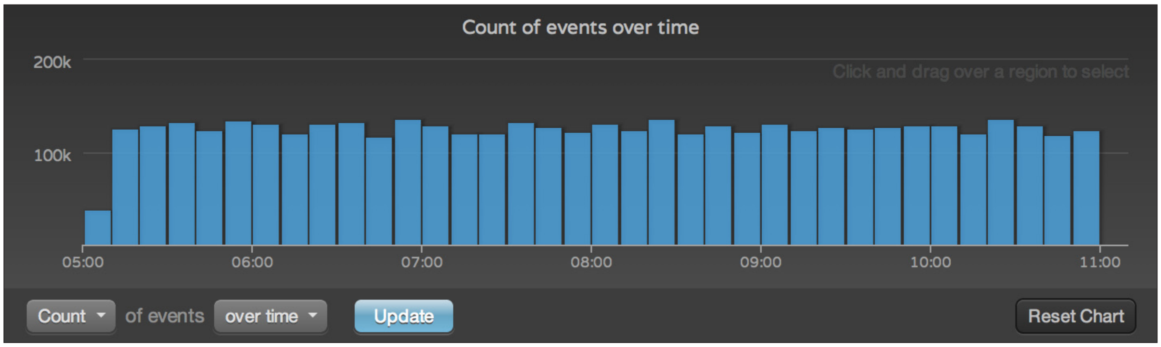


Figure 14. An example of a bar chart using count of events over time.

If the count function is used in conjunction with a single field grouping instead of time series, then a bar chart is created with quantities listed from greatest to least.

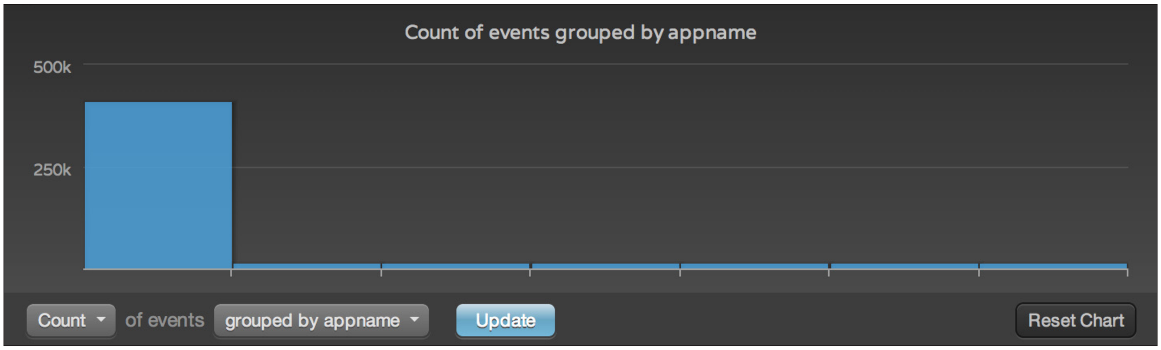


Figure 15. An example of a bar chart using count of events grouped by a field.

Line Charts

All functions, except the count function, are mathematical and require a field to apply the equation against. When performing a mathematical function on a field and grouping by time series, a line chart is created.

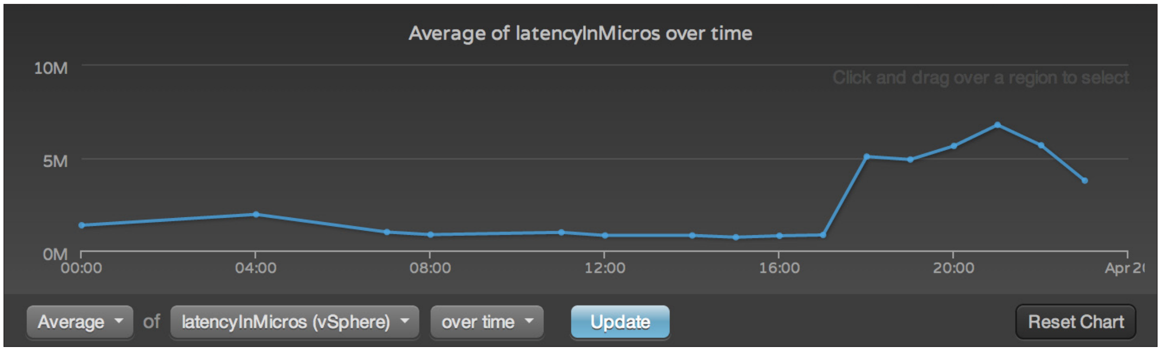

Average of latencyInMicros over time
Click and drag over a region to select
10M
5M
0M
00:00
04:00
08:00
12:00
16:00
20:00
Apr 2(
Average
of
latencyInMicros (vSphere)
over time
Update
Reset Chart

Figure 16. An example of a line chart using average of a field over time.

Stacked Charts

By default, the overview chart on the Interactive Analytics page of Log Insight is a count of events over time. If one field is added to the time series grouping, then a stacked chart is created.

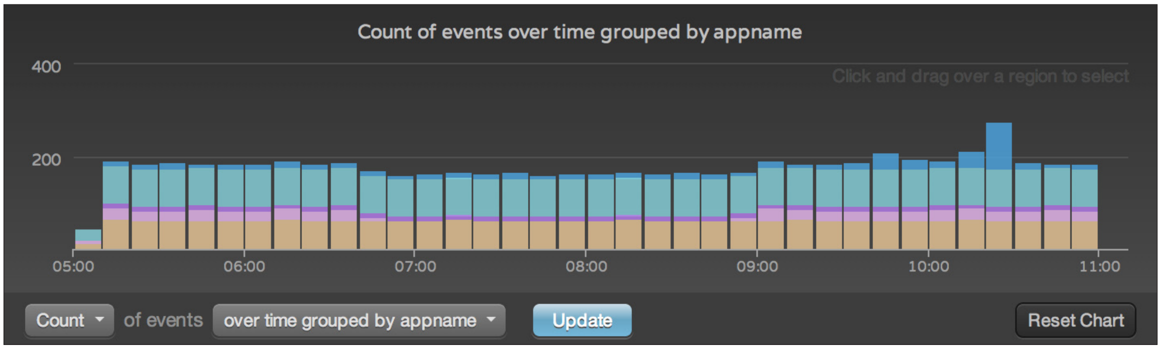

Count of events over time grouped by appname
Click and drag over a region to select
400
200
05:00
06:00
07:00
08:00
09:00
10:00
11:00
Count
of events
over time grouped by appname
Update
Reset Chart

Figure 17. An example of a stacked bar chart using count of events over time with a field.

If grouping by time series plus a field and any function except count is used, a stacked line chart is created.

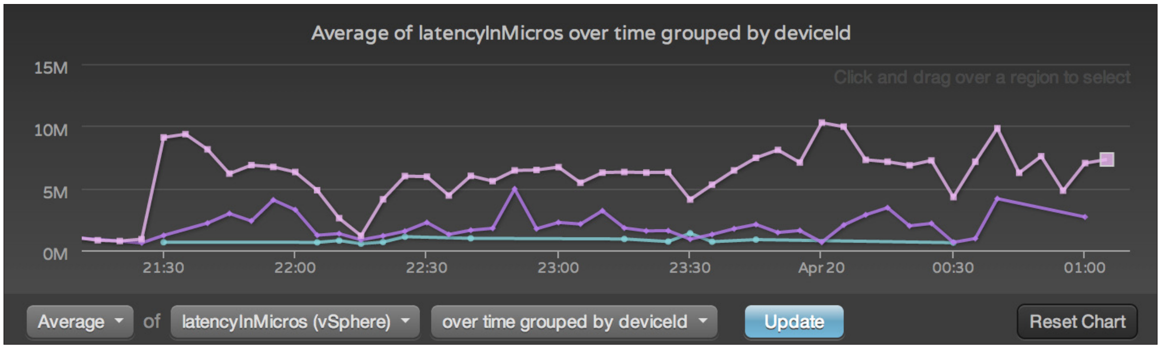

Average of latencyInMicros over time grouped by deviceId
15M
10M
5M
0M
Click and drag over a region to select
21:30
22:00
22:30
23:00
23:30
Apr 20
00:30
01:00
Average
of
latencyInMicros (vSphere)
over time grouped by deviceId
Update
Reset Chart

Figure 18. An example of a stacked line chart using average of a field over time grouped by a field.

Stacked charts are powerful when attempting to find anomalies for an object. Consideration needs to be taken based on the number of objects that could be returned. In general, the following best practices apply:

- If the number of objects per bar returned will be <10, then stacked charts are encouraged.

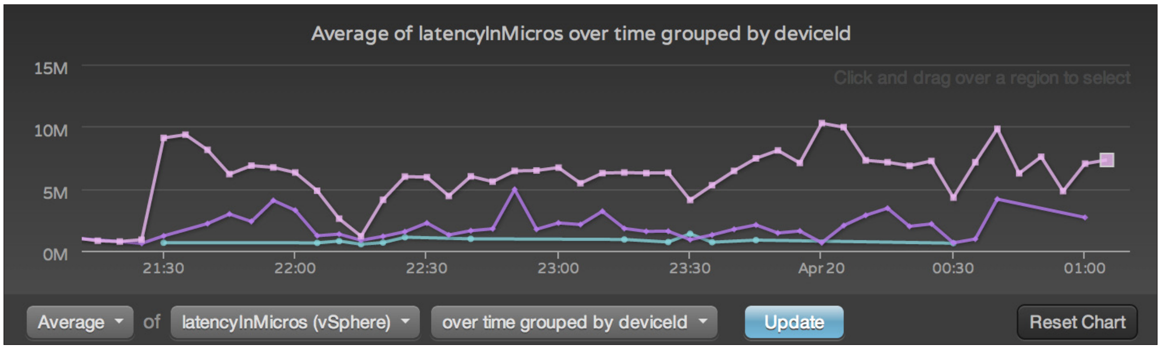

Average of latencyInMicros over time grouped by deviceId
15M
10M
5M
0M
Click and drag over a region to select
21:30
22:00
22:30
23:00
23:30
Apr 20
00:30
01:00
Average
of
latencyInMicros (vSphere)
over time grouped by deviceId
Update
Reset Chart

Figure 19. An example of a stacked line chart with a small number of objects. The chart is easy to read and understand.

- If the number of objects returned per bar is or could be 10-20, then stacked charts can be good, but consideration must be taken when visually representing the chart in a content pack.
- If the number of objects returned per bar is or could be >20, then stacked charts are discouraged.

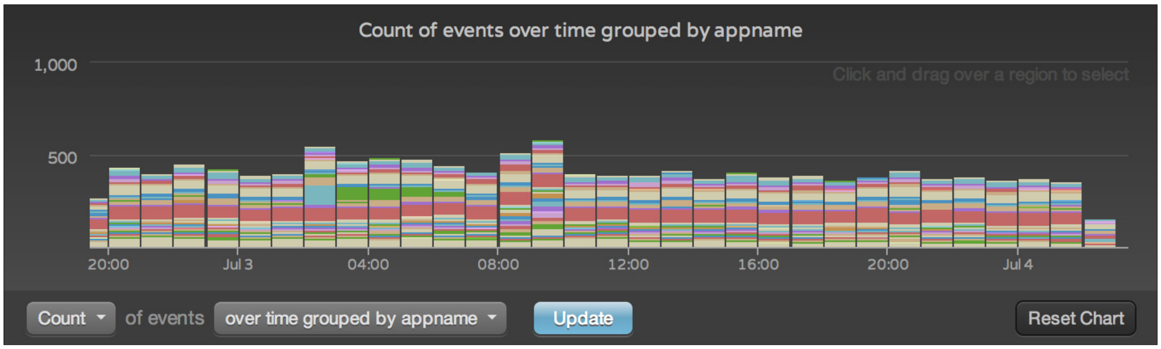

Count of events over time grouped by appname
1,000
500
Click and drag over a region to select
20:00
Jul 3
04:00
08:00
12:00
16:00
20:00
Jul 4
Count
of events
over time grouped by appname
Update
Reset Chart

Figure 20. An example of a stacked bar chart with a large number of objects. The chart is hard to read and understand.

The reason for the above recommendations is because more objects mean more resources necessary to parse and display information. In addition, there are a fixed number of colors, 9 in the current version, and distinguishing between objects may become challenging depending on the number of objects returned.

Multi-Colored Charts

If a grouping is created using more than one field and time series, then a multi-colored chart is created. The chart will consist of two colors that interchange. Each interchange represents a new time range. Multi-colored charts can be hard to interpret so the value of such a chart should be considered before including it in a content pack.

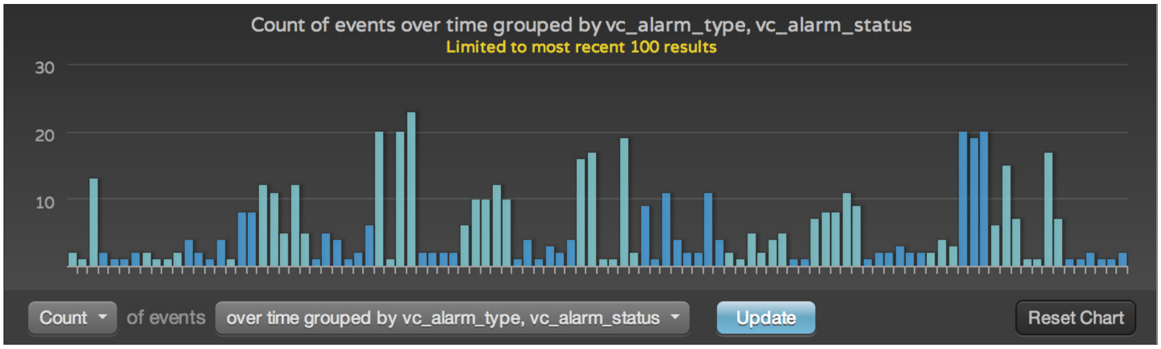

Count of events over time grouped by vc_alarm_type, vc_alarm_status
Limited to most recent 100 results
30
20
10
Count
of events
over time grouped by vc_alarm_type, vc_alarm_status
Update
Reset Chart

Figure 21. An example of a multi-colored bar chart using count of events over time grouped by two fields.

When grouping by multiple fields, consider removing the time series for a more easily understood bar chart.

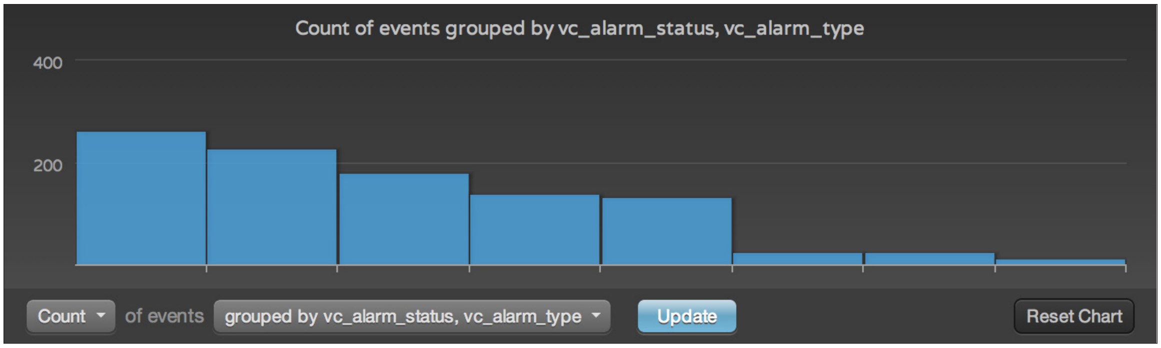

Count of events grouped by vc_alarm_status, vc_alarm_type
400
200
Count
of events
grouped by vc_alarm_status, vc_alarm_type
Update
Reset Chart

Figure 22. An example of a multi-field grouping bar chart using count of events grouped by two fields.

If multiple fields are important given a time range, then multiple charts could be created for each field individually over the time range. The charts could then be displayed in the same column of a dashboard group in a content pack.

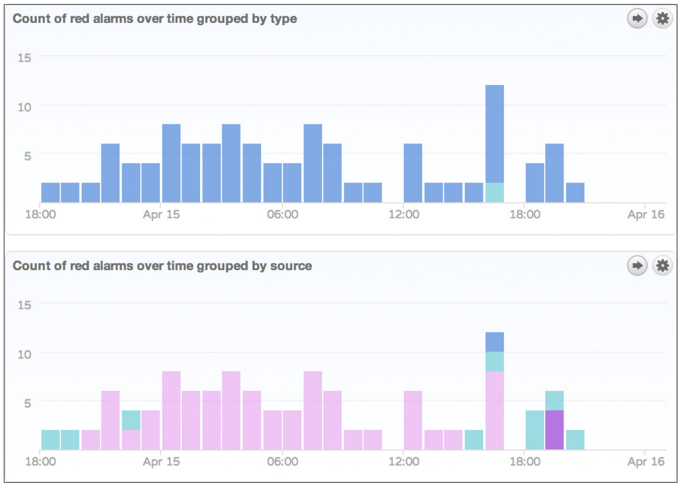
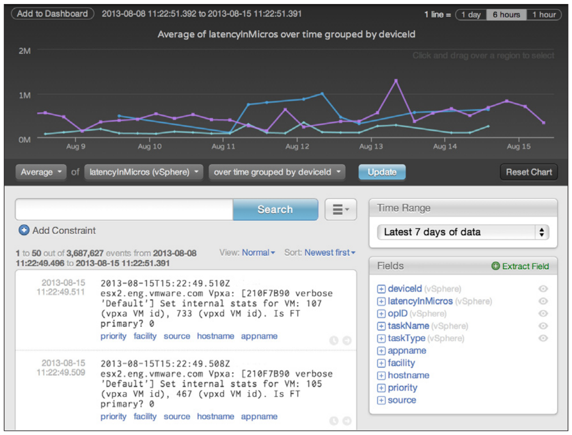

Count of red alarms over time grouped by type
15
10
5
18:00
Apr 15
06:00
12:00
18:00
Apr 16
Count of red alarms over time grouped by source
15
10
5
18:00
Apr 15
06:00
12:00
18:00
Apr 16

Figure 23. An example of two similar charts stacked. Notice how one red alarm in blue matches mostly pink sources.

Message Queries

When constructing an aggregation query, the message query should only return results relevant to the aggregation query. This makes analyzing easier and ensures only relevant fields are shown.



2013-08-08 11:22:51.392 to 2013-08-15 11:22:51.391

Average of latencyInMicros over time grouped by deviceId

Average of latencyInMicros (vSphere) over time grouped by deviceId

Time Range: Latest 7 days of data

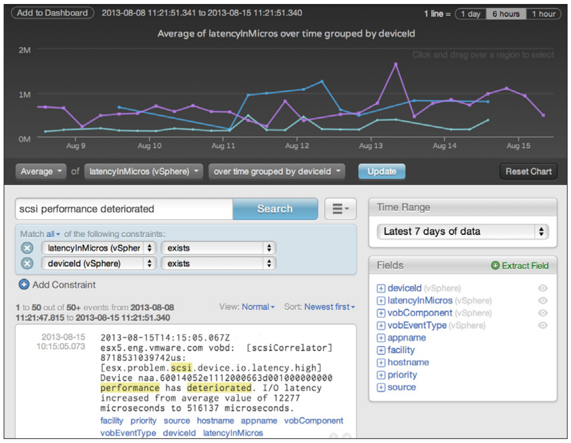
1 to 50 out of 3,687,627 events from 2013-08-08 11:22:49.496 to 2013-08-15 11:22:51.391 View: Normal Sort: Newest first

```
2013-08-15    2013-08-15T15:22:49.510Z
11:22:49.511  esx2.eng.vmware.com Vpxa: [210F7B90 verbose
              'Default'] Set internal stats for VM: 107
              (vpxa VM id), 733 (vpxd VM id). Is FT
              primary? 0
              priority facility source hostname appname

2013-08-15    2013-08-15T15:22:49.508Z
11:22:49.509  esx2.eng.vmware.com Vpxa: [210F7B90 verbose
              'Default'] Set internal stats for VM: 105
              (vpxa VM id), 467 (vpxd VM id). Is FT
              primary? 0
              priority facility source hostname appname
```

Fields: deviceId (vSphere), latencyInMicros (vSphere), opID (vSphere), taskName (vSphere), taskType (vSphere), appname, facility, hostname, priority, source

Figure 24. An example of an aggregation query without a message query. This is not recommended.



2013-08-08 11:21:51.341 to 2013-08-15 11:21:51.340

Average of latencyInMicros over time grouped by deviceId

Average of latencyInMicros (vSphere) over time grouped by deviceId

Search: scsi performance deteriorated

Match all of the following constraints:
latencyInMicros (vSpher exists
deviceId (vSphere) exists

Time Range: Latest 7 days of data

1 to 50 out of 50+ events from 2013-08-08 11:21:47.815 to 2013-08-15 11:21:51.340 View: Normal Sort: Newest first

```
2013-08-15    2013-08-15T14:15:05.067Z
10:15:05.073  esx5.eng.vmware.com vobd:  [scsiCorrelator]
              8718531039742us:
              [esx.problem.scsi.device.io.latency.high]
              Device naa.60014052e1112000663d001000000000
              performance has deteriorated. I/O latency
              increased from average value of 12277
              microseconds to 516137 microseconds.
              facility priority source hostname appname vobComponent
              vobEventType deviceId latencyInMicros
```

Fields: deviceId (vSphere), latencyInMicros (vSphere), vobComponent (vSphere), vobEventType (vSphere), appname, facility, hostname, priority, source

Figure 25. An example of an aggregation query with a message query. This is recommended.

Alerts

Alerts provide a way to trigger a reaction when a certain type of event is seen. Log Insight supports two different types of alerts natively:

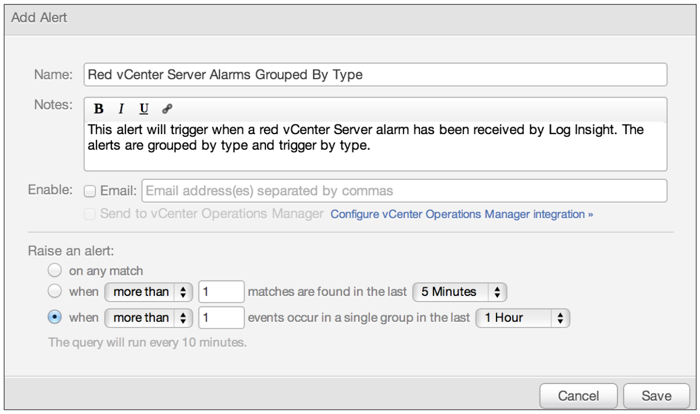
- Email
- vCenter Operations Manager

Alerts can only be saved in user space and as such, all content pack alerts are disabled by default. If an enabled alert is created and then exported as part of a content pack, then the alert will be disabled in the content pack. This means that email and/or vCenter Operations Manager settings are not contained and cannot be added to a content pack.

Thresholds

It is important to understand how thresholds work to ensure that, if enabled, a content pack alert does not unintentionally spam a user. When considering a threshold, there are two things to keep in mind:

- **How frequently to trigger the alert:** Log Insight comes with pre-defined frequencies. Important: Alerts will only trigger once for a given threshold window.
- **How often to check if an alert state has occurred:** An alert is triggered by a query. Alerts, just like queries, are not real-time in the current version. For each threshold window, a pre-determined query frequency has been allocated. Changing the threshold will change the query time.



Add Alert

Name: Red vCenter Server Alarms Grouped By Type

Notes: This alert will trigger when a red vCenter Server alarm has been received by Log Insight. The alerts are grouped by type and trigger by type.

Enable: □ Email: Email address(es) separated by commas
□ Send to vCenter Operations Manager

Raise an alert:
○ on any match
○ when more than 1 matches are found in the last 5 Minutes
● when more than 1 events occur in a single group in the last 1 Hour

The query will run every 10 minutes.

Figure 26. An example of an alert. The threshold has been set to trigger when one red vCenter Server alarm for any type is seen in the last hour. The query runs every 10 minutes and if the alert triggers, it will not run again for one hour.

Dashboards

Dashboard Groups

A content pack is made up of one or more dashboard pages known as dashboard groups.

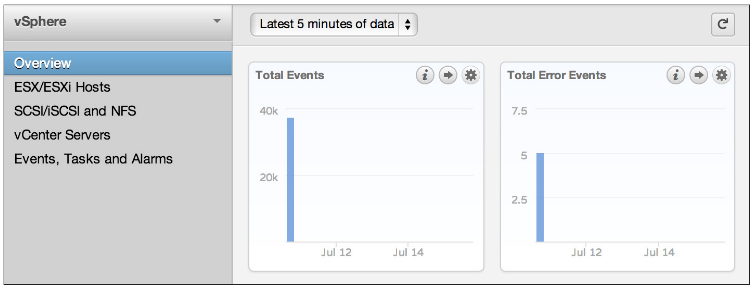


Figure 27. The vSphere content pack. In the left navigation bar, below the name of the content pack, are the dashboard groups.

When creating dashboard groups, the following best practices apply:

- Content packs commonly contain a minimum of 3 dashboard groups. The best practice is to start with an overview dashboard group to provide high-level information about the events for a particular product or application. In addition to the overview dashboard group, dashboard groups should be created based on logical groupings of events. The logical groupings are product-specific or application-specific, but some common approaches are: performance, faults, and auditing. It is also common to create dashboard groups per component like disk and controller. With the component approach, it is important to note that it is only effective if queries can be constructed to return results from specific components. If this is not possible, then the logical approach is recommended.
- When naming dashboard groups, make the title generic and avoid adding product-specific or application-specific names unless being used in a component specific fashion. For example, in the vSphere content pack, the dashboard groups are called *ESX/ESXi hosts* and *SCSi/iSCSI and NFS* instead of *VMware ESX/ESXi hosts* and *VMware SCSI/iSCSI and NFS*.
- A dashboard group should contain a minimum of 3 dashboard widgets and a maximum of 8 dashboard widgets. With any less than 3 dashboard widgets the amount of knowledge that can be attained by the dashboard group is minimal. In addition, having a lot of dashboard groups with only a limited amount of dashboard widgets requires a user to switch between different pages and does not provide information in a coherent way. Conversely, any more than 8 dashboard widgets per dashboard group can result in the following:
  - Too much information: A user may not know where to start or what is most important.
  - Resource intensive: Each widget is a query that needs to be run against the system.

  When nearing or exceeding 8 dashboard widgets in a dashboard group, separate information and create multiple dashboard groups. If a dashboard widget is applicable to one or more dashboard groups, it is recommended to create the widget in each applicable dashboard group.

Dashboard Widgets

There are two different types of dashboard widgets in the current version of Log Insight:

- **Chart:** contains a visual representation of events with a link to a saved query.
- **Query:** contains title links to saved queries.

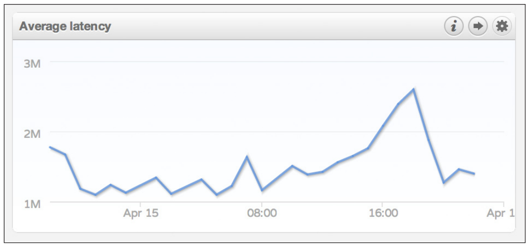
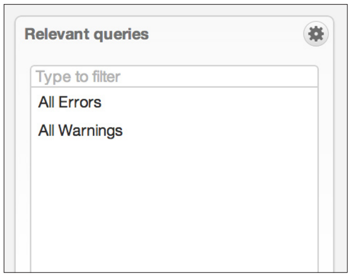


Figure 28. An example of a chart widget.


Relevant queries

Type to filter

All Errors

All Warnings

Figure 29. An example of a query widget.

Chart

A dashboard chart widget contains a visual representation of events. A chart can either be represented as a bar or line chart and either can be displayed in a stacked fashion. The following best practices apply:

- Charts can contain a lot of information so avoid having more than two chart widgets per row. In some rare cases, three chart widgets can be used effectively, but more than three is strongly discouraged. When determining whether chart widgets are readable or not, be sure to use the minimum resolution supported by Log Insight (1024 x 768) as one cannot assume that users of the product will have a better resolution.

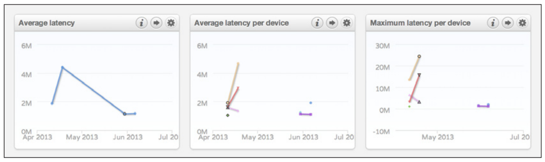


Figure 30. An example of three chart widgets in the same row. With more content, these widgets may become hard to read.

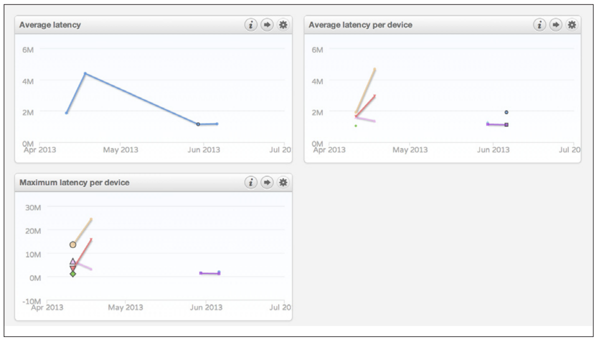


Figure 31. An example of two chart widgets per row. Even with more content these charts should be more readable.

- If any row except the last row has a single chart widget, then it is recommended to make that widget full-width.

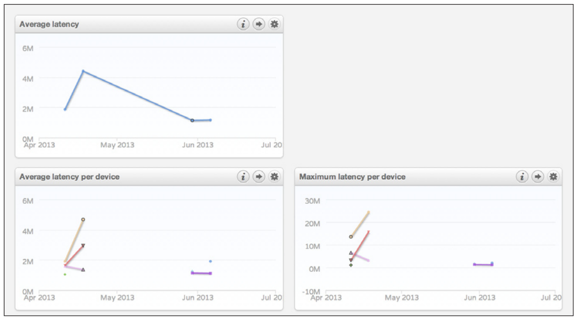


Figure 32. An example with a half-width chart on the top row. This is not recommended.

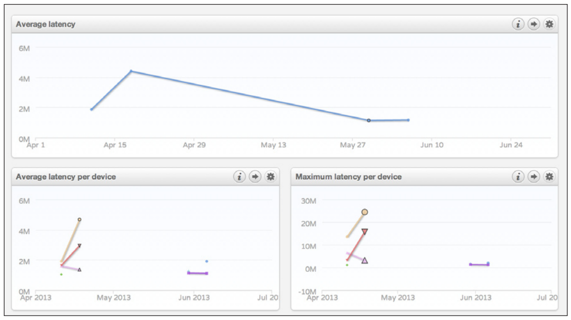


Figure 33. An example of a full-width chart on the top row. This is recommended.

- When naming a chart widget, use a descriptive title and avoid cryptic field names. For example, an extracted field is called *vmw_error_message*. Instead of calling a chart *Count of vmw_error_message*, call it *Count of error messages.*
- Similar charts can be saved and stacked in the same column of a dashboard group for visual comparison. Examples of such charts include:
  - Average X of events over time + Maximum X of events over time. Given the different functions used, it is possible that the Y-axis of the charts will not be the same scale.
  - Count of events over time grouped by X + Count of events over time grouped by Y.

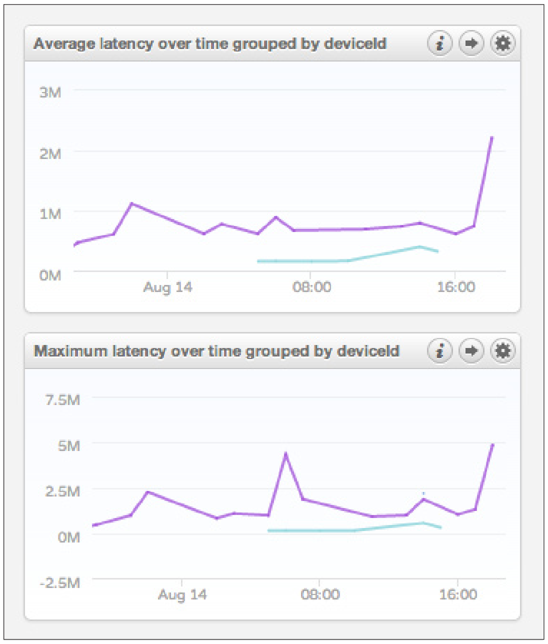

Average latency over time grouped by deviceId
3M
2M
1M
0M
Aug 14
08:00
16:00
Maximum latency over time grouped by deviceId
7.5M
5M
2.5M
0M
-2.5M
Aug 14
08:00
16:00

Figure 34. An example of two similar charts using different functions stacked. Notice the scale of the charts does not match.

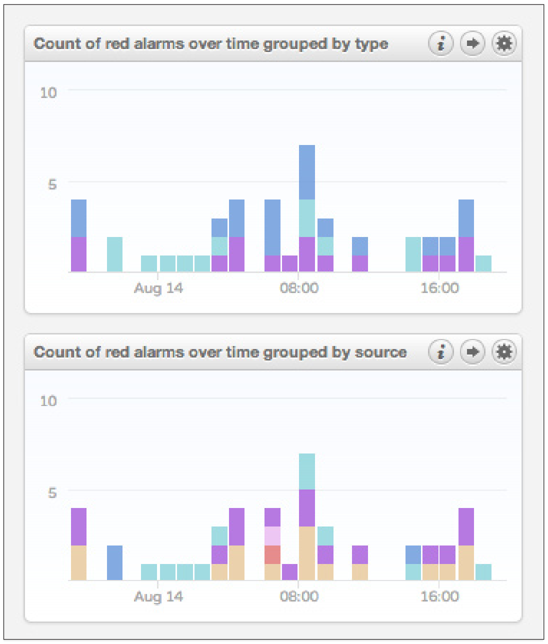
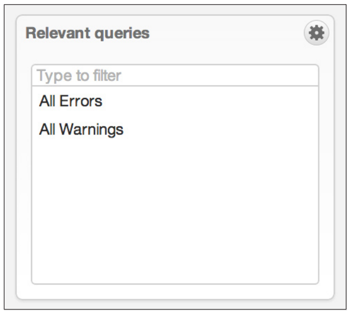

Count of red alarms over time grouped by type
10
5
Aug 14
08:00
16:00
Count of red alarms over time grouped by source
10
5
Aug 14
08:00
16:00

Figure 35. An example of two similar charts using different groupings stacked. For this type of query, the scale of the charts will match.

Query

A dashboard query widget contains a title that is a link to a pre-defined query. Query widgets are often used when a chart widget does not provide significant value, but a query does. Query widgets cannot be created from the web UI of Log Insight and as such it is not recommended that they be used in content packs in the current version.



All Errors
All Warnings

Figure 36. An example of a query widget.

Widgets

Widgets can be modified in a variety of ways including:

- **Rename:** To rename a widget, select the name of the widget. When naming a chart widget, use a descriptive title and avoid cryptic field names.
- **Resize:** To resize a widget, hover over the right edge of a widget's contents.

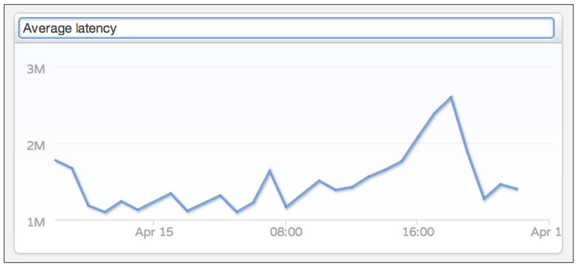


Figure 37. Renaming a widget.

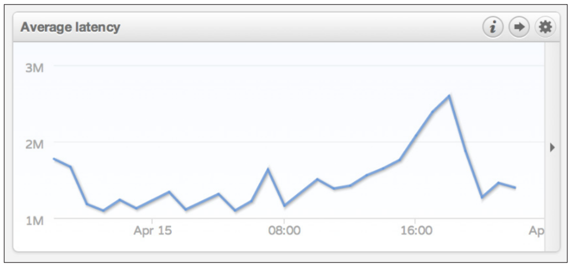


Figure 38. Resizing a widget.

- **Move**:
  - **Within a dashboard group:** To move a widget within a dashboard group, select between the title and the action buttons and drag to the new location. Important: It is not possible to create a new row between two existing rows.

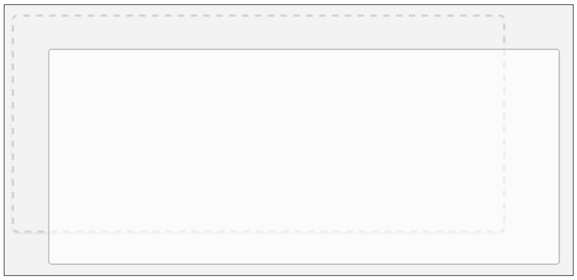


Figure 39. Moving a widget within a dashboard group.

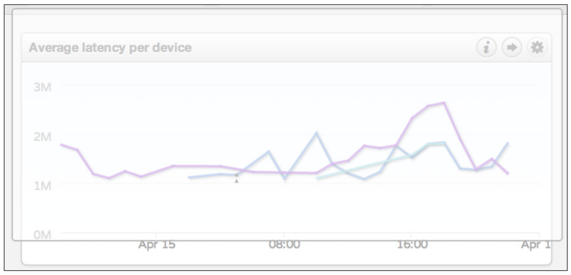


Figure 40. Attempting to add a new row in a dashboard group.

Instead, move the widget to the left-most position of the row below the row desired and move all widgets after the new widget down.

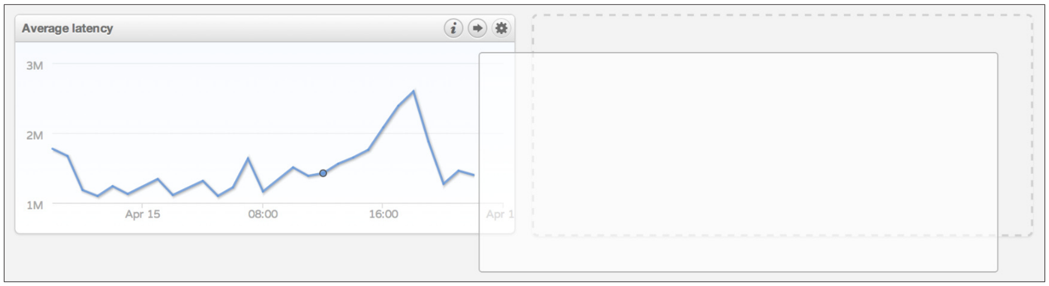


Figure 41. Moving a widget to create a new row in a dashboard group.

  - **Between dashboard groups:** To move a widget between dashboard groups, select the gear action button followed by *Move to Dashboard*.

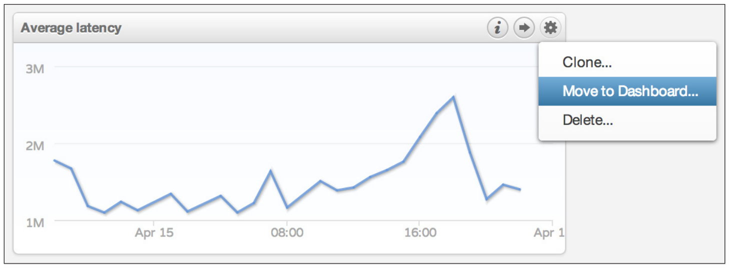


Figure 42. Moving a widget to a new dashboard group.

- **Clone**: To clone a widget, select the gear action button followed by *Clone*. Important: When cloning a chart widget, any fields the chart relies on are not cloned. Instead, cloned chart widget fields are defined by the cloned source. For this reason, cloned widgets should not be used in content packs as they may cause content packs to be dependent on other content packs.

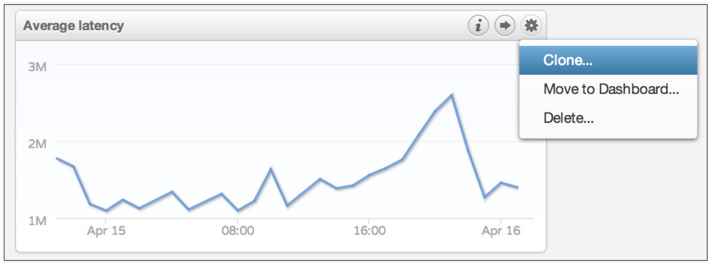


Figure 43. Cloning a widget.

- **Edit Information:** To edit the notes section of a widget, select the i button followed by *Edit*.

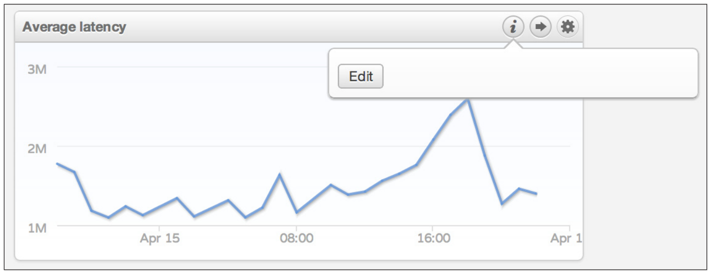


Figure 44. Editing the information field of a widget.

The notes section is very important and should be populated for every dashboard widget. Information added can be text, a link to documentation, a knowledge base article, or a forum. Information provided should answer the following questions:

– Why is this widget important?

– What is a “good” / “bad” value?

– Where can more information be obtained?

The underlying query for a widget cannot be modified. In order to change the underlying query, a new widget needs to be created and the old widget needs to be deleted. For chart widgets, the directions are:

- Go to the widget on the Dashboards page.
- Select the *Open in Interactive Analytics* arrow button within the widget.
- Modify the query as desired.
- Select the *Add to Dashboard* button on the Interactive Analytics page.
- Select the *Delete* button from the gear button within the old widget from the Dashboards page.

Content Packs

With an understanding of what makes up a content pack and the best practices when performing each operation, it is now time to view, export, import, edit, and publish the content.

View

To view saved content:

- Navigate to the *Content Packs* section by selecting the gear icon in the navigation bar and selecting Content Packs.

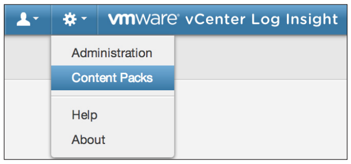
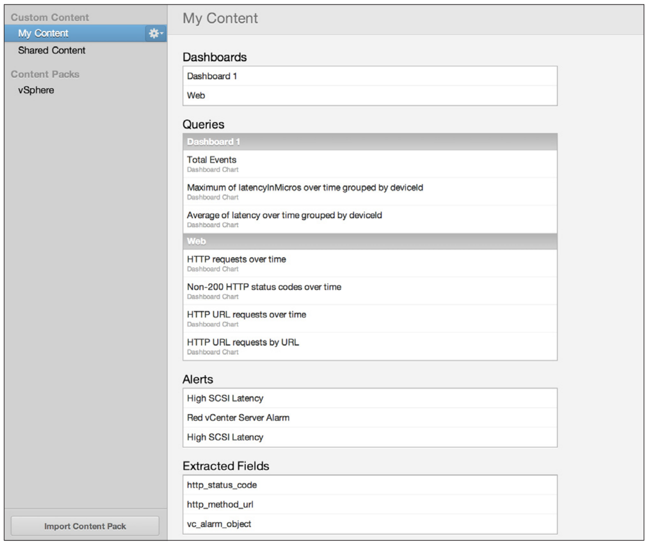


Figure 45. Content Packs menu option. Important: The Administration option will only be visible to Admin users.

- Select where the content was saved. For content pack authors, content is saved under *Custom Content* and, if following the best practices in the Getting Started section of this document, saved content will appear under *My Content*.



Custom Content
My Content
Shared Content
Content Packs
vSphere
Import Content Pack

My Content

Dashboards

Dashboard 1
Web

Queries

Dashboard 1
Total Events
Dashboard Chart
Maximum of latencyInMicros over time grouped by deviceId
Dashboard Chart
Average of latency over time grouped by deviceId
Dashboard Chart
Web
HTTP requests over time
Dashboard Chart
Non-200 HTTP status codes over time
Dashboard Chart
HTTP URL requests over time
Dashboard Chart
HTTP URL requests by URL
Dashboard Chart

Alerts

High SCSI Latency
Red vCenter Server Alarm
High SCSI Latency

Extracted Fields

http_status_code
http_method_url
vc_alarm_object

Figure 46. An example of saved content.

In general, a content pack should have:

- 3+ Dashboards (dashboard groups)
- 3+ Queries (chart widgets) per dashboard (9+ total)
- 5+ Alerts
- 20+ Extracted Fields

Export

Private

To export all information saved in a dashboard for private consumption:

- Select the content and then select the gear icon to the right of the dashboard name.
- Select the *Export Content Pack* link.
- Give the content pack a name. The recommended format is: <*Company*> – <*Product*> v<*Version*> (e.g. VMware – vSphere v1.0). Ideally, the content pack name should be under 30 characters to prevent word wrapping.
- Select *Export*.

Once done, a file ending in VLCP, which stands for **v**Center **L**og Insight **C**ontent **P**ack, will be downloaded.

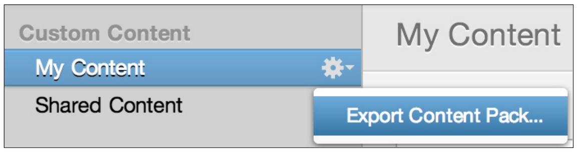
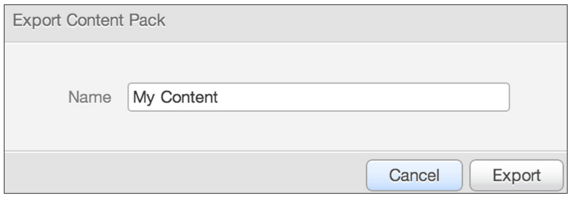


Figure 47. How to export a content pack.



Name: My Content

Figure 48. Export content pack dialog box.

Public

To export all information saved in a dashboard for public consumption:

- Select the content and then select the gear icon to the right of the dashboard name.
- Select the shift key and then select the *Export Content Pack with Namespace...* link.
- Give the content pack a name. The recommended format is: <*Company*> – <*Product*> v<*Version*> (e.g. VMware – vSphere v1.0).
- Give the content pack a namespace. The recommended format is: <*Ext*>.<*Domain*>.<*Product*> (e.g. com.vmware.vsphere).
- Select *Export*.

Once done, a file ending in VLCP, which stands for **v**Center **L**og Insight **C**ontent **P**ack, will be downloaded.

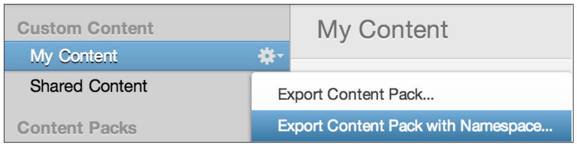
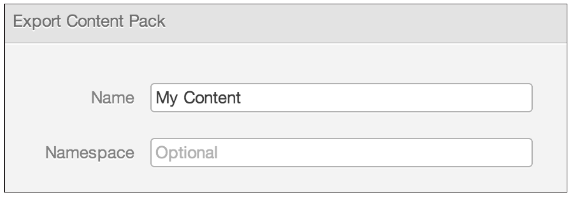


Figure 49. How to export a content pack with namespace.



Name: My Content
Namespace: Optional

Figure 50. Export content pack with namespace dialog box.

Import

To import a content pack:

- Select the *Import Content Pack* link in the bottom of the left navigation bar.
- Select *Browse*… to specify the location of the VLCP file.
- Select *Import*.

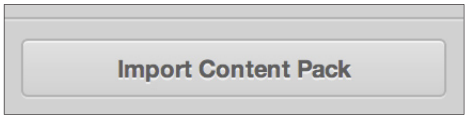
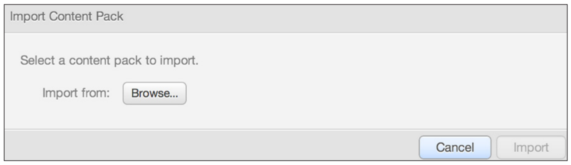


Figure 51. Import button.


Import Content Pack

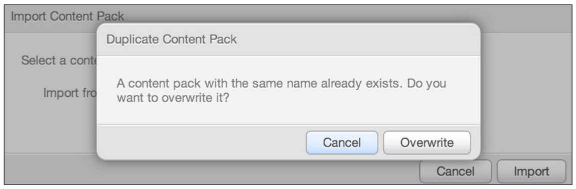
Select a content pack to import.

Import from: Browse...

Figure 52. Importing a content pack dialog box.

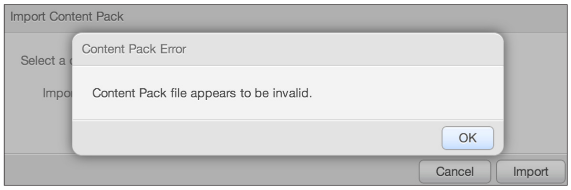
When importing a content pack, some warning/error events can occur. These include:

- **Duplicate Name:** A Duplicate Name means that another content pack is installed in the system that has the same unique identifier. In this case, the options are to either choose *Overwrite* to replace the existing content pack or *Cancel* to keep the existing content pack.
- **Invalid Format:** Invalid Format means that the VLCP file was manually edited and contains syntax errors. The syntax errors need to be fixed before the content pack can be imported. As VLCP files should not be manually edited, there is no easy way to locate and fix syntax errors.


Duplicate Content Pack

A content pack with the same name already exists. Do you want to overwrite it?

Figure 53. Duplicate content pack warning dialog box.


Content Pack Error

Content Pack file appears to be invalid.

Figure 54. Content pack error dialog box.

Edit

As imported content packs are immutable, or read-only, content packs should be edited from the instance of Log Insight they were created on. It is possible to clone content pack dashboard widgets to custom dashboards, also known as user space, however the cloning operation does not clone extracted fields, which may result in one content pack relying on a second content pack to function properly. In such a situation, without the second content pack being imported, the first content pack would contain orphaned fields. As such, the recommendation is to modify content packs on the instance of Log Insight used to create the original content pack. This means the original Log Insight instance should be properly backed up. If the original Log Insight instance is not available, the content pack should be created from scratch if modification to the content pack is required.

Publish

Once a content pack has been created, it can be published on the Log Insight marketplace, which is located on VMware Solution Exchange. The requirements for content pack publishing are as follows:

- **Content pack:** A VLCP file.
- **Events:** Appropriate events necessary to validate content pack.
- **Demo/Story:** Example of how the content pack brings value (e.g. YouTube video).
- **Documentation:** How to configure the product/application to forward logs to Log Insight.

For more information, see the Resources section below.

Conclusions

Content packs are a powerful way to extend the knowledge contained within Log Insight. When creating a content pack, several best practices must be considered as outlined below.

Getting Started

Instance

- The instance of Log Insight used to create a content pack must be backed up. If the instance used to create a content pack becomes unusable, the content pack must be recreated on a different instance in order to modify.
- Do not attempt to edit a content pack from a different instance of Log Insight than the one that created the content pack unless the intention is to recreate the content pack.

User

- Use a separate content pack author user on Log Insight for every content pack being created.

Queries

Message Queries

- Use keyword queries whenever possible.
- If keyword queries are not sufficient use globs.
- Use regular expressions only if keywords and globs are not sufficient. When using regular expressions, provide as many keywords as possible.
- Make queries as specific as possible. Content pack queries should only match events applicable to the product/application for which the content pack was designed.

Field Extraction

- Minimize the amount of regular expressions used whenever possible.
- Confirm that the regular expression value will match every applicable log message.
- Provide as much pre-keyword and/or post-keyword context as possible.
- When naming a field:
  - Use the following naming standard: <prefix>_<field>_<name>
  - Use underscore instead of space.
  - Use all lowercase letters.
  - <prefix> = something applicable to the content pack.
- Test to validate the extracted field is working as expected.

Aggregation Queries

- When grouping by time series, do not add more than one field.
- Do not group by time series and one field if the number of unique fields is or could be over 20.
- When grouping by more than one field and time series, ensure the time series adds value.
- If the time series is important for more than one field, consider creating individual charts per field and per time series and save charts in the same column of a dashboard group.
- When constructing aggregation queries, ensure message queries return equivalent results.

Alerts Queries

- Create alerts primarily for critical events.
- Limit alerts using thresholds. In general, the user should not get more than 6 alerts per hour.
- Any saved alerts will be disabled once exported as part of a content pack. Email and/or vCenter Operations Manager definitions will not be included in a content pack.
- Be sure to enter descriptive information about the alert so the user will know why it is important.

Dashboards

Dashboard Groups

- Consider starting with an overview dashboard group.
- Create dashboard groups based on a specific type of message (e.g. overview, performance, etc.) and not based on a specific type of component (e.g. compute, network, storage).
- It is recommended to duplicate the same dashboard widget in multiple dashboards groups if the dashboard widget is applicable in each dashboard group.
- Target at least 3 dashboard groups in a content pack.
- Dashboard groups cannot be reordered, but dashboard widgets can be moved.

Dashboard Widgets

- Target at least 3 dashboard widgets per dashboard group.
- Do not put more than 3 dashboard widgets in the same row.
- Do not put more than 8 dashboard widgets in a dashboard group.
- When displaying similar information in different formats, ensure each format brings value.
- Stack related dashboards together for easier viewing.
- Give the dashboard widgets descriptive names. Do not use field names in widget titles.
- Include notes for every dashboard widget. Ensure the notes answer questions such as, *"Why is the widget important?"* and *"Where can additional information be found?"*
- Changing the definition of a field requires that all dashboard widgets created with the previous field definition be re-created to take advantage of the new field definition.
- The query definition of a dashboard widget cannot be modified. Instead, a new widget needs to be created and the old widget needs to be removed.

Content Packs

- A content pack should contain a minimum of 3 dashboards, 9 total widgets, 5 alerts, and 20 fields.
- When exporting a content pack use the naming format: <*Company*> – <*Product*> *v*<*Version*>.
  Ideally, the content pack name should be under 30 characters to prevent word wrapping.
- When exporting a content pack for publishing, export with a namespace.
- When exporting with a namespace use the namespace format: <*Ext*>.<*Domain*>.<*Product*>.

Resources

More information about Log Insight and Log Insight content packs can be found using the links below.

- **VMware vCenter Log Insight documentation:**
  http://www.vmware.com/support/pubs/log-insight-pubs.html
- **VMware vCenter Log Insight communities:**
  http://communities.vmware.com/community/vmtn/vcenter/vcenter-log-insight
- **VMware vCenter Log Insight marketplace:**
  https://solutionexchange.vmware.com/store/loginsight
- **VMware vCenter Log Insight ideas:**
  http://loginsight.vmware.com

Acknowledgments

The author would like to thank the following individuals for their technical review of this white paper:
Jon Herlocker, Spiros Xanthos, Chengdu Huang, Chris Blinn, Pavan Sura, and Yogita Patil.

About the Author

Steve Flanders is a Senior Solutions Architect at VMware. He has an extensive background in designing and implementing cloud solutions with focuses on ensuring scalability and promoting a cloud vision. Steve has helped architect several cloud offerings including VMware's Cloud Foundry, ATT's Synaptic Storage as a Service, and EMC's Atmos Online. Steve is the author of SFlanders.net, a technology-centric weblog focusing on a variety of topics including cloud computing, virtualization, and system administration.

- Follow Steve's blog at http://sflanders.net
- Follow Steve on Twitter: @smflanders

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com