White Paper

Analytics in Content QC

Analyze-Measure-Optimize!

Interra Systems, Inc.
1601 S. De Anza Boulevard, Suite 212
Cupertino, CA 95014
USA

1. INTRODUCTION

As the media industry shifted from tape based workflows to file-based workflows, file-based automated content QC solutions started providing significant cost and reliability advantages compared to the traditional QC by visual inspection. Over the last decade, file-based media workflows have become much more sophisticated as the expectations and flexibilities both on the content contribution as well as distribution side have increased. This has put tremendous pressure of ensuring content quality on those who create and deliver the content.

File-based content QC solutions have steadily evolved to handle the emerging challenges in identifying and validating content quality issues in these sophisticated environments. They have become a trusted and indispensable component of modern file-based workflows for any quality conscious media organization.

With continuous use of content QC solutions, a large amount of QC data gets generated. It is possible to analyze this data and identify trends which help in deeper understanding of content quality in an organization. Such a deeper understanding can help in making strategic improvements in the content QC processes and achieve greater organizational efficiencies. This paper describes a *Measure > Analyze > Optimize* framework for incorporating data analytics in content QC processes. The framework is applied in different contexts to show how to get the best out of content QC.

A core management principle is: anything that can be measured can be improved further. Or in other words, if you can't measure something, you can't improve it. This old adage is equally applicable to content QC processes also. Traditional file-based QC solutions have focused on a simple model of ***C**heck a file* → ***R**eview its verification report* → ***T**ake necessary actions* → ***F**orget*. While this model works well for day to day QC operations, it isn't effective for long term planning of content QC processes and making strategic changes. Better tools are needed which can give a more holistic view and deeper insights about how the content QC has happened over a long period of time and across different departments / sites. At Interra Systems, we look at the overall strategic content QC planning and execution in a MAO framework [Figure 1]. MAO stands for: ***M**easure* → ***A**nalyze* → ***O**ptimize*.

Figure 1 The MAO Framework

Specifically

- At the *Measure* stage, appropriate tools are used to track important QC performance metrics over long term
- At the *Analyze* stage, the metrics are analyzed to identify common patterns and operational issues; the analysis gives pointers to areas of improvement and suggests required changes
- At the *Optimize* stage, the changes based on the analysis are applied to the content QC processes
- The cycle is repeated:
  - We *measure* again to verify that the applied changes have led to improvements in different QC performance metrics
  - New measurements are *analyzed* for new issues and required changes

In the following, we look at several examples where the MAO framework can be used to bring improvements in content QC processes. While we look at concrete examples with specific metadata fields or errors, the ideas are easily generalized for any metadata field or error type.

2. ASSET CATEGORIZATION

A content rich organization has huge number of assets at its disposal. At times, the organization has pretty vague idea of what all types of assets it has acquired or created over the years. Many a times, there is no proper content management system (CMS) to keep track of all assets. It may either not be affordable, or the effort required to catalog all assets may be long or expensive. While a CMS may not be around, usually all the assets pass through some level of QC. QC reports contain huge amount of metadata information as well as error information about the files. The database of QC reports can be exploited to explore and answer a number of interesting questions. For example:

- How many hours of content have been processed?
- How many files have been processed over a specific span of time?
- What are the different types of content processed (format, resolution, bit rate, etc.)?
- What percentage of files have black bars?
- What is the distribution of aspect ratio over all files?
- What are the bit rates being used in different files?
- Which are my HD files? Across all assets? In a particular watch folder?
- What are the most frequent fatal errors?
- Which files were expected to be HD but were not really HD?
- Which files didn't have proper aspect ratio?
- Which files had dropouts in them?

Figure 2 presents an example situation. A total number of 38,864 files had undergone QC. 24 different values of resolution were found in these files. The most common resolution was 625(SD). About 60% of files were encoded at this resolution. Second most common value was 1080(HD) which covered about another 34% of files. There was one odd file with a resolution of 528x480.

Total Occurences	38864
Number of distinct values	24
Most frequent value	625SD(720x576)
Least frequent value	528x480

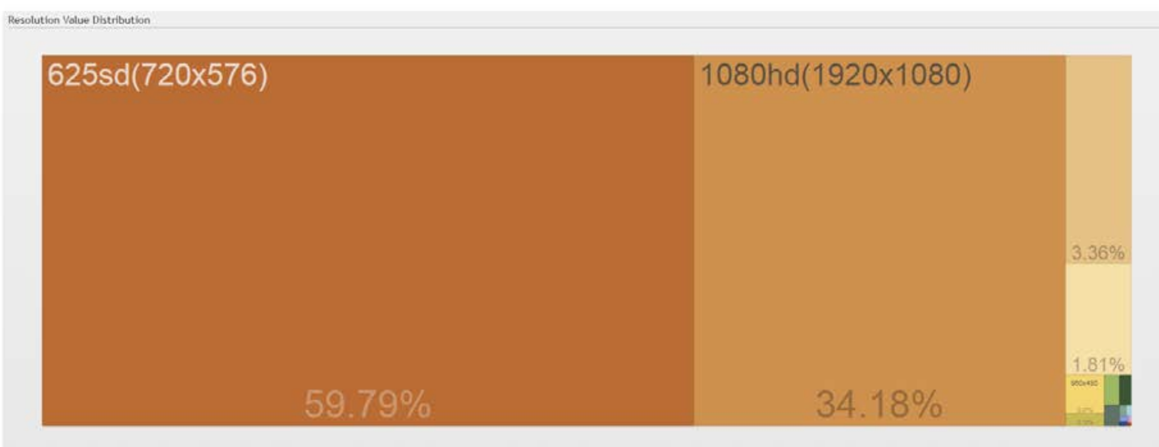

Figure 2 Resolution of different files

The data is useful in multiple ways:

- If the organization is making a transition from SD to HD content, then it can get a sense of amount of legacy content it still carries with it. This helps in planning for other resources in the organization. For example, as long as significant amount of SD assets are still lying in the organization, one may need to continue maintaining an SD-to-HD upconversion workflow.
- One can cross check why they are still having some assets with unique resolutions like 528x480 mentioned above. Upon review, they can clean up such files from their system. Most probably they are the result of one odd transcode to that specific resolution.

3. QC RESULTS SUMMARIZATION

An in-depth analysis of QC results is imperative for answering a number of important questions related to content QC process. Here are some examples:

- What percentage of files are failing due to some error?
- What are the different kinds of errors present in different files?
- Which errors are more prominent than others?
- How the errors are distributed for different content providers or different workflows?
- Which stages in the workflow are typical culprits in introducing errors?
- How the failure rate has changed over time? Has it improved or not?
- During what period of the year, more failures tend to occur?
- If any of the QC tasks have been aborted? What were the reasons?

Let us look at some examples. Figure 3 shows a high level summary of what percentage of files are failing the QC checks.

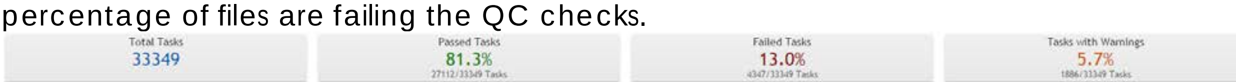

Figure 3 High level summary of task results

In general the goal in improving content quality is to ensure that less and less number of files are failing. One may need to dig deeper in the data to see how the failure rate is changing over time. Figure 4 shows how the number of tasks (in success, failure or warnings) are changing from month to month. In this particular example, one doesn't see much improvement in the failure rate with time.

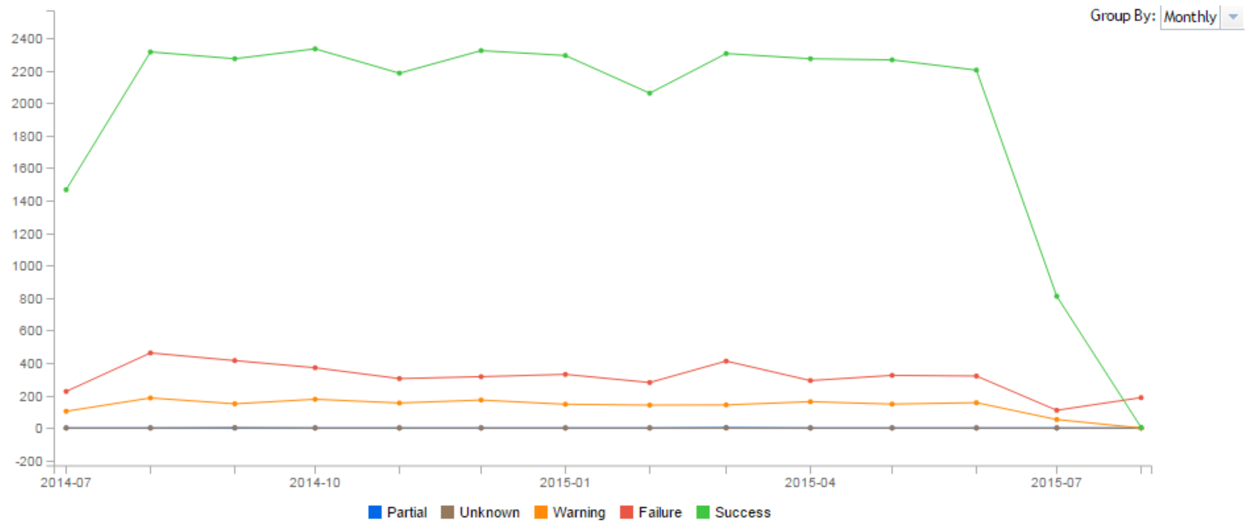

Figure 4 Month wise variation in success and failures

Looking at this data, the goal of an organization is to identify the reasons why the failure rate is not improving with time. Once the reasons have been identified, the next goal is to carry out the operational improvements and achieve decreasing failure rates.

IDENTIFYING GOOD AND BAD PERFORMERS

There are several ways to slice the failure rate data:

- One can restrict it to specific watch folders to identify folders which have more problems
- One can look at the same data for different types of content separately (e.g. HD vs. SD content, Ingest vs. Playout content)
- If there is a multi-site installation, one may want to see which site tends to have more failures than others

Figure 5 shows distribution of failure rates over multiple sites. One can see that the Sydney office has much better success rate than others. Mumbai office is the worst offender where all the files are failing. Performance of New York center is in between.

Figure 5 Task results by QC sites

One could use several strategies to leverage this information in optimizing overall content quality in organization. Here are a couple:

- Understand the best practices followed in centers having higher content quality and disseminate them across the rest of the organization
- Create a competitive spirit amongst different centers by rewarding the ones which are performing better and penalizing the ones which are not showing any signs of improvement on year by year basis

We can also slice the failure rate data over parameters like Test Plans, Watch Folders, Content Locations, Checkers etc. See Figure 6 for an example.

Figure 6 Task results by different criteria

Amongst the watch folders, we can see that *Stories* watch folder seems to have higher failure rate than others. Amongst the test plans, we note that *SD Open Stories Test Plan* tends to have more failures than other test plans. These data points let us have clear action plans of where we should focus our attention to improve content quality.

The checker wise distribution is more useful for seeing if QC tasks are evenly distributed across different checkers or not. In this particular example it seems that two checkers are overloaded and are handling most of the tasks.

TYPICAL ERRORS

Another interesting view point is to look at the most common errors found across all tasks. Figure 7 shows the most common errors in terms of number of files affected with the particular error.

Errors

Scheduled Tasks				
33349	27112	4347	1886	

Error	Tasks with Error		Tasks with Error Ignored	
Video Quality : Chroma Change	5401	16.20%	22	0.41%
Video : Chroma Format	4870	14.60%	25	0.51%
Video Quality : RGB Color Gamut	3667	11.00%	0	
Video Quality : Pixelation	3005	9.01%	0	
Video Quality : Blockiness	2878	8.63%	0	
Video Quality : Video Flash	2381	7.14%	0	
Audio : Channels	2152	6.45%	0	
Video : Display Aspect Ratio	2132	6.39%	14	0.66%
Container : Audio Tracks Count	2039	6.11%	7	0.34%
Container : Video Tracks Count	1860	5.58%	1	0.05%
Video : Resolution	1844	5.53%	0	
Video : Display Format	1800	5.40%	0	
Audio Quality : Noise:Wow and Flutter	1738	5.21%	0	
Audio Quality : Loudness (CALM Act)	1413	4.24%	0	
Audio : Bit Rate Change	1301	3.90%	0	
Audio : Bits per Sample	1043	3.13%	0	
Video : Luma Bits Per Pixel	1009	3.03%	0	
Video : Average Bit Rate	972	2.91%	0	
Video Quality : Blurriness	817	2.45%	1	0.12%
Video : Picture Scanning Type	699	2.10%	0	
Audio Quality : Loudness (EBU)	695	2.08%	0	
Video Quality : Dropout	230	0.69%	0	
Video Quality : White Point	160	0.48%	36	22.50%
Audio Quality : Watermark:Nielsen	97	0.29%	0	

Figure 7 Common errors affecting different tasks

We can see that chroma change error has been the most common error affecting about 16.2% of the files in this case. Some of the more important problems plaguing our content are RGB color gamut, pixelation, blockiness etc. The action plan in this case is to look at the files with specific problems in detail and identifying if there are common causes in the workflow which are leading to the problem recurring in so many files.

Sometimes, this data may not be sufficient. It may miss out on specific errors which are happening in fewer files but the number of occurrences of the error in those files is very high. Figure 8 looks at the total number of occurrences of errors across all tasks. You may notice that the order of errors has actually changed in the two tables.

▾ Video Quality

Metadata	Errors	Ignored Errors	Severity
Chroma Change	21755	38	21507
Pixelation	6788	0	6788
RGB Color Gamut	5517	0	5036
Blockiness	5498	0	5498
Video Flash	3336	0	1666 1670
Blurriness	835	9	826
Dropout	748	0	534 214
White Point	744	104	640 104
Video Signal Level	523	0	523
Dropout:Block Error	201	0	201
Freeze Frames	69	0	69
Motion Jerk	63	0	63
Black Bars	61	0	61
Black Frames	61	3	58 3
Brightness	44	0	44
Contrast	31	0	31
Freeze Frames (Presence)	27	0	27
Color Bar	22	0	22
Video Noise	15	7	8 7
Field Order:Source	14	0	14
Cadence	8	0	8
Stripe Error	1	0	1

Figure 8 Total errors of specific type across all tasks

4. CAPACITY PLANNING

An important aspect of having an efficient QC process is to ensure that the QC system is lean and mean. Several important goals need to be addressed:

- Checkers shouldn't be sitting idle
- QC tasks should be completed in reasonable amount of time
- Higher priority tasks should be given resources accordingly
- Enough capacity should be there to handle peak load situations
- In case of multi-site QC systems, the resources should be equitably distributed
- Regular review of distribution of resources and their usage should be done

Some of these goals are conflicting with each other. For example, if the organization gets hyper-active during specific seasons, then naturally peak load will be much higher in that period. If more checkers are bought to handle the peak load, during the rest of the year they are likely to be sitting idle. In this section, we discuss several metrics and their applications to achieve these goals.

ESSENTIAL PERFORMANCE CRITERIA

Some of the essential performance criteria are listed in Figure 9. File duration is the length of the particular content in HH:MM:SS format. Task duration is the amount of time taken to verify the file. Performance index is the ratio of File Duration with Task Duration. e.g. 2x means that QC of a one hour file gets completed in half an hour. The higher the performance index the better. Finally, the QC rate tells us how many files were processed each day on average. This also includes the time for which the QC system was sitting idle.

Figure 9 Essential performance criteria

An important goal here is to increase the average performance index. There are several ways to improve performance index:

- Allocate more cores per task
- Disable non-essential checks from the test plan
- Ensure faster access to content from checkers

While throwing more CPUs at the QC task may seem appealing, there are some comments in order. Adding more CPUs doesn't improve performance proportionally. For example, for an SD file, using 32 cores per task may not give substantial improvements over using 8 cores per task although 8 cores per task would be about 6 times faster than 1 core per task. Further, while increasing performance index may reduce the QC time, the system may be sitting idle if there is not enough content to be processed all the time. Adding more checkers to a QC system costs money. Thus, a reasonable target performance index should be chosen as a good balance between the need to get QC done fast and ensuring that QC system is reasonably utilized all the time. In this context, it is useful to slice the performance data on various parameters:

- We may want to look at performance index for SD/HD/4K/8K files separately - this helps us decide which files require more cores to be thrown at it to achieve better overall performance
- We may want to look at performance index for different watch folders separately - we may want to throw more cores at more important watch folders
- We may want to look at the performance index for long form content and short form content / advertisements separately; while we may focus on improving the performance index for high value long term content, we may ignore others

Each check in a test plan adds to overall QC time. In particular, video/audio quality checks have significant contributions to the QC time. A review of test plan wise QC results can tell us about the checks which never fail in a particular workflow. It doesn't mean that we should simply switch off the checks which never fail. Some of these checks may be a MUST from regulatory compliance perspective and can never be switched off. But for others, a decision can be taken after review over a longer term if such checks should be disabled altogether from the test plan.

Sometimes access of files from storage area to QC checkers becomes a bottleneck in overall QC time. If such a situation is identified, one should focus on improving file access time.

Figure 10 provides a deeper insight into the distribution of performance index over different files. It is easier to spot files for which the performance is way too slow. One can then take a more focused approach to improving performance of these files (if deemed necessary).

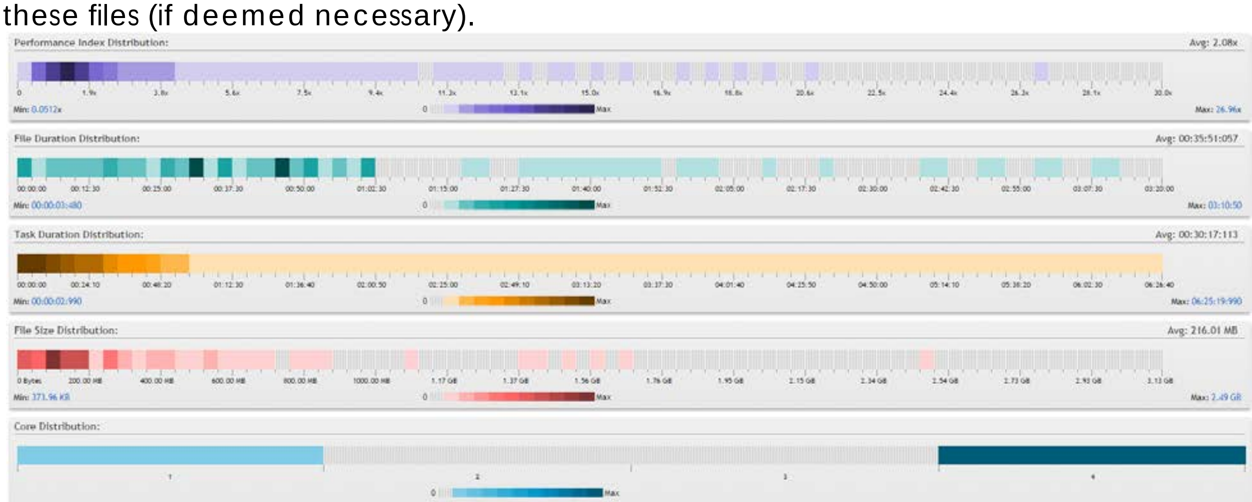

Figure 10 Distribution of performance criteria over all tasks

Figure 11 shows several other performance parameters. The first chart shows the percentage of time a QC system was in use vs. the time it was sitting idle (doing nothing). Ideally, the system should never be sitting idle, but such a dream can be achieved only if the system has a steady stream of incoming tasks (which is rarely the case). The second chart shows the distribution of number of cores used during the time the system was active. In this particular example, it shows that only 4 cores were being used during 42 percent of the time. The peak usage of 36 cores (probably the number of licensed cores) was rare. Thus, it looks like that total number of licensed cores is on the higher side. The third chart shows that 96% of the time, either there were no tasks in the system, or a task got started as soon as it was put in the system. Only for 4% of the time, tasks had to wait for a while. There is also a separate statistic of average/maximum wait time which tells us the amount of time a task has to wait in queue for before it gets started. A good system would have a low average wait time and a low maximum wait time. The queue size shouldn't grow too high. At the same time, the system shouldn't be sitting idle most of the time. The fourth chart shows how the queue size varied over the time tasks were in queue.

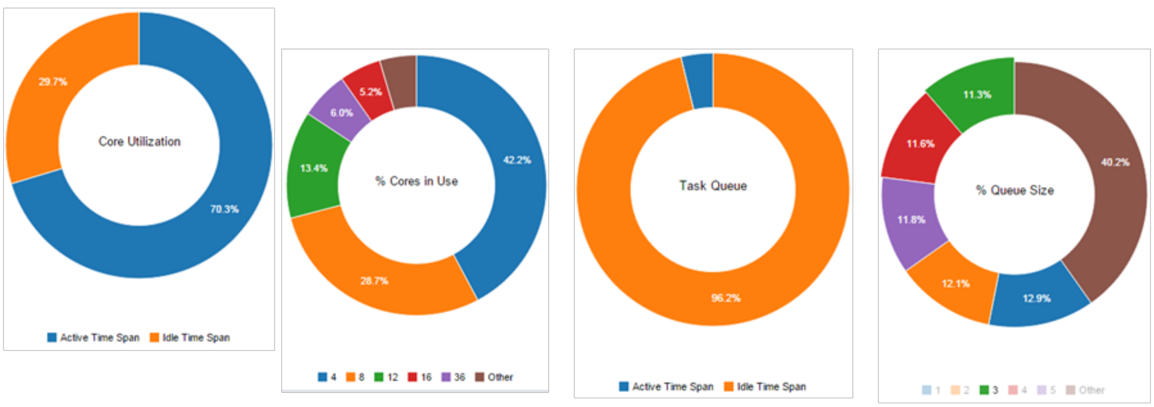

Figure 11 Core Utilization and Task Queue

Sometimes it is useful to do a detailed hour by hour or day by day analysis of how the queue size or the core utilization varied over time. Figure 12 shows how (maximum) queue size varied over time. Figure 13 shows how (maximum) the core utilization varied over time. It is easy to spot the month (August) in which queue size was highest, and the day in which it maxed out. These detailed timelines can help in identifying if some specific events caused massive increases in resource utilization or requirements.

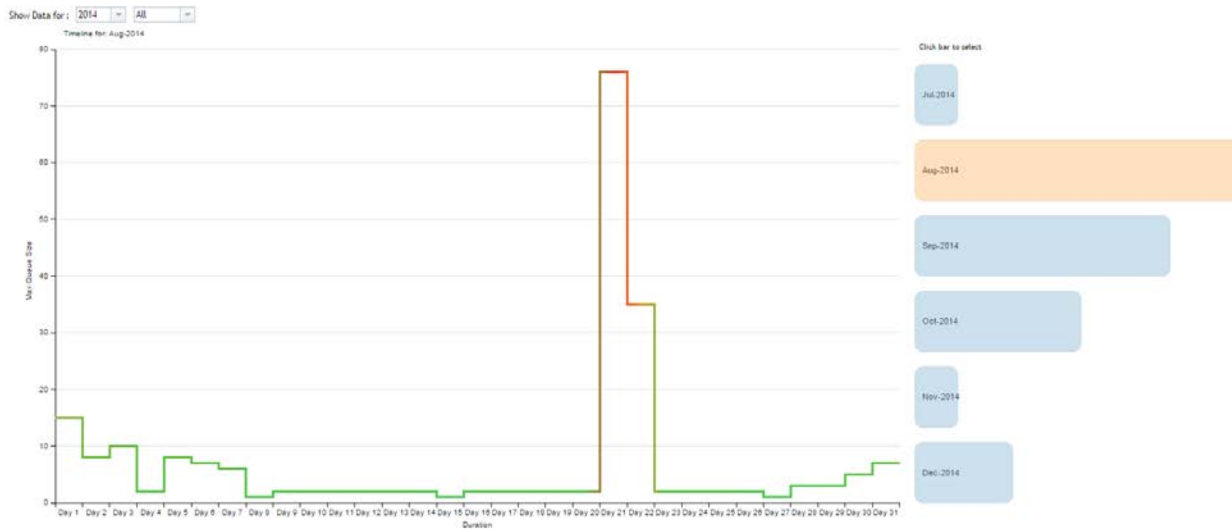

Figure 12 Queue Size History

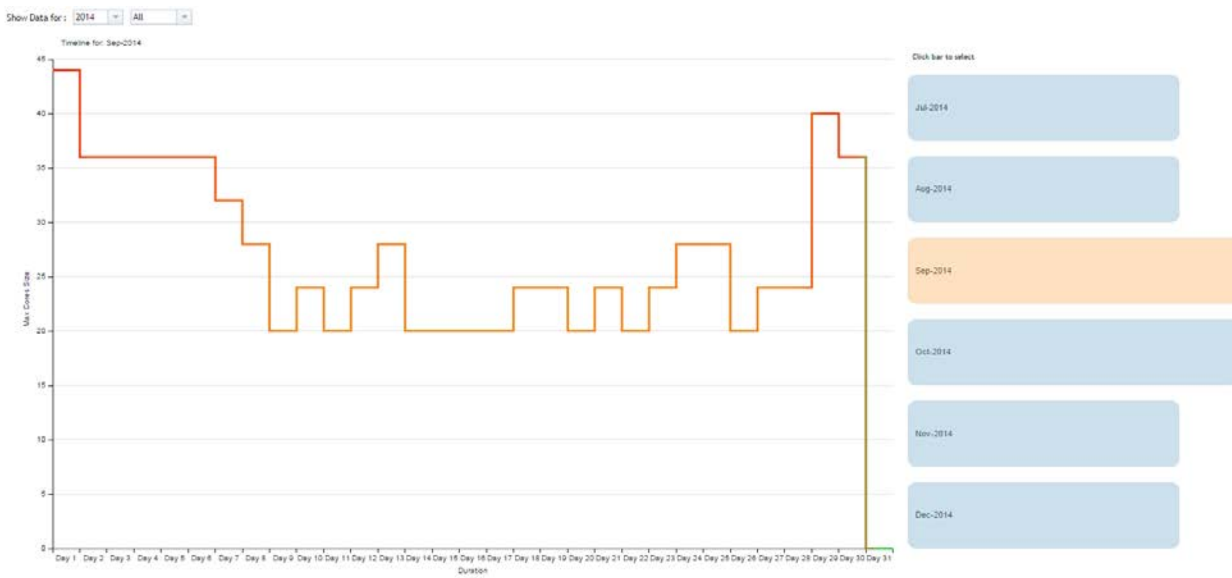

Figure 13 Core Utilization History

INTER-DEPARTMENTAL AND MULTI-SITE RESOURCE SHARING

Sometimes different departments of the same organization decide to buy their independent copies of QC systems. In the case of multiple offices in different locations, having independent QC systems cannot be avoided at all. The analytics system can import data from these different installations and provide a combined top level overview for strategic planning. Some of the important questions are:

- Which sites/departments are using all their licensed cores effectively? Which sites are just sitting on their licensed cores?
- How the overall performance index is varying for different sites? Which sites are more resource hungry? Do we need to make more budgetary

allocations for specific sites (depending on their actual need and organizational priorities)?

Figure 14 shows an example of (average) core utilization across multiple sites. It is clear that the Amsterdam site is utilizing its licensed cores capacity quite well. The Sydney site seems to be wasting valuable QC resources. One may choose to move some of the cores from the Sydney site to Amsterdam site for overall improvement in QC performance.

Figure 14 Site Wise Core Utilization

Figure 15 shows comparison of performance index, queue size, waiting time, and QC rate for the four sites. It is clear that Amsterdam site has much higher queue size and waiting time. Its QC rate is also higher. It does make sense to move some of the QC resources from other sites (say Mumbai and Sydney) to Amsterdam.

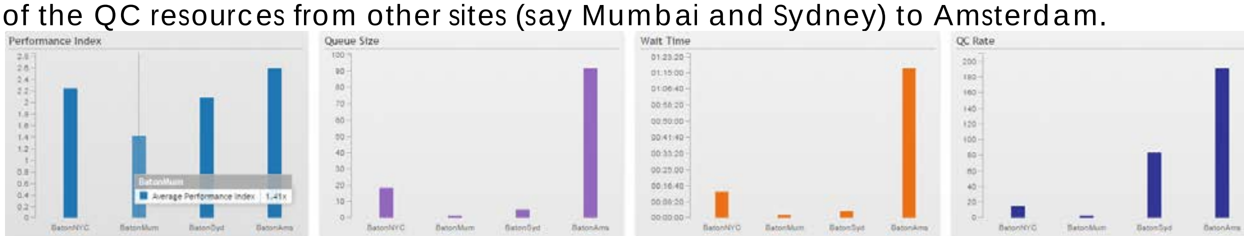

Figure 15 Site Wise Performance

5. CONCLUDING REMARKS

The holy grail of media organizations is to provide a pervasive, immersive as well as *glitch-free* experience to *happy customers.* The life-cycle of content has become much more complicated due to the large number of options available in both content contribution and distribution. Structured and automated content QC has become an integral part of the modern file-based media workflows towards assisting content developers in delivering high quality content experience. Analytics is a young and growing area in file-based QC and its full potential is not yet realized. We hope to see more action in this area in coming years.

6. THE INTERRA SYSTEMS SOLUTION

BATON+™ FOR CONSISTENT QUALITY, DATA ANALYTICS, MANAGED WORKFLOW ACROSS THE ENTERPRISE

Interra Systems' Baton+™ is an advanced enterprise-class audio/video content analytics and content-centric workflow solution, which seamlessly integrates with multiple Baton™ systems managing, analyzing and reporting Baton’s long-term

QC data across multiple Baton installations to improve workflow efficiency. Baton+ helps in making strategic improvements in the content QC processes and achieve greater organizational efficiencies. Click here for more: http://www.interrasystems.com/baton+.php

7. ABOUT INTERRA SYSTEMS

Interra Systems provides software-based QC, monitoring, and analysis solutions to the digital media industry. The company's solutions include Baton, a market-leading, enterprise-class QC solution that automatically ensures media content readiness; Orion, a real-time content monitoring solution that simplifies the delivery of superior quality video; and Vega, a family of audio/video analyzers for standards compliance, debug, and interoperability of encoded streams.

For more information, please visit http://www.interrasystems.com.

