FASAMS USER EXPERIENCE AND TECHNICAL REVIEW AND FINDINGS

Final Version 1.3

Financial and Services Accountability Management Systems User Experience and Technical Review Findings Document

Abstract

This document will present high-level review and findings of the DCF FASAMS system focusing on User Experience and underlying Technology implementation. In this document the key issues will be highlighted along with the future directions.

DIGITAL CONSULTING WIZARDS

Contents

Table of Figures

Signature and Acceptance Page

I have reviewed this FASAMS Technical and UX findings document and agree that the content of the document is accurate, and clearly describes the findings for the project.

Signature: _____ Date: _____

Print Name: _____

Title: _____

Agency: _____

Role: _____

Signature: _____ Date: _____

Print Name: _____

Title: _____

Agency: _____

Role: _____

Signature: _____ Date: _____

Print Name: _____

Title: _____

Agency: _____

Role: _____

1. Revision History

Version	Date	Author	Revision Description
1.0	6/28/2019	OZ	Final Deliverable
1.1	07/17/2019	OZ	Amendment - Source code review
1.2	8/1/19	OZ	Amendment – Source code review
1.3	**8/7/19**	**OZ**	**Amendment – AWS review**

2. Executive Summary

2.1.Summary

At its core, the FASAMS System was developed to answer one (five-part) question:

> Which **clients** received what **services**, from which **providers**, at what **cost**, for what **outcomes**?

The system was constructed to have the seven Managing Entities (MEs) upload (on a monthly basis) the information about the managed providers who are delivering services in their geography to clients – the providers themselves do not upload information directly into FASAMS. Once the data is uploaded by the MEs into FASAMS, a series of reports are available to the DCF team.

Upon launch (January 1, 2019) DCF quickly started experiencing issues with the system, including the speed of the reporting module, which seemed to be getting worse as more data was uploaded by the MEs.

The OZ team was asked to review the FASAMS application and did the following:

- Interviewed various DCF employees, a provider and several MEs.
- Technical review of architecture, database, source code (limited due to lack of access).
- Reviewed the documentation, used the application and ran reports to perform a user experience review.
- Submitted an initial assessment document on May 9, 2019.
- Performed a high-level quality review.
- Completed the review and documented detailed findings and recommendations.

While the initial consensus (based on the Tallahassee DCF interviews and the OZ team team's initial review) was that the system was not usable in its current state, and while we have not seen evidence that the fundamental five-part question mentioned above can be answered with an acceptable level of confidence, we believe the core of the system can be remediated to do so.

To get the system to work, we believe that the following critical remediation activities need to be performed prior to any further enhancements or modifications to the FASAMS system (please see Critical Remediation Activities section below for details):

1. Perform a full Quality Assurance (QA) test of the system.
2. Overhaul the database.
3. Fix the submission process.
4. Create a new, separate reporting database (data warehouse).
5. Fix the existing reports.

6. Fix the overall user experience of FASAMS.

There are other, specific or detailed improvements that we recommend throughout the document below, but the items in the list above are the most critical, at a higher level (see Technical and User Experience sections).

2.2.FASAMS Future Vision / Opportunities

Once the above issues are remediated, DCF can improve its quality, timeliness of services available and value through innovative digital technologies such as Process Automation, Artificial Intelligence and Advanced Analytics , Robotic Process Automation, Interactive Dashboards, Cloud Services, Mobile Applications and Big Data to strengthen its priority of improving the child welfare system of care statewide. Below are some examples of what the future of FASAMS could be:

Artificial Intelligence (AI)

Artificial intelligence and advanced analytics solutions can assist with predicting and optimizing health and welfare outcomes, anticipating demand for services, identifying waste, fraud and abuse, and reducing payment errors.

Robotics Process Automation (RPA)

RPA can simplify the FASAMS data load process and reduce the number of errors. A simplified process will reduce the MEs processing burden (time, effort) and improve satisfaction. In addition, the new process will lead to improved data quality.

Interactive Dashboards

Interactive dashboards can provide insights and allow different stakeholders to focus on outcomes. Dashboards may be used to answer questions such as: What treatments are providing best results for the cost? What is the average cost to treat? Beyond simple reporting, interactive dashboards can allow slicing and drill down on dimensions such as: time, regions, MEs, providers, ailments, procedure categories, outcomes, cost.

Allow Providers to Enter Information Directly

Creating a way for providers to enter information into FASAMS directly, bypassing the MEs, would streamline the submission process by eliminating the need for the MEs to perform the monthly batch processes. Provider satisfaction would increase, because they would be able to see the information that they submit, something they have been requesting for years.

Cloud Services/Mobile Applications

Cloud services can extend the FASAMS application to a native mobile platform powered by intelligent agents/bots (AI). This approach will provide a direct channel of communication between providers and DCF. The mobile platform may also be used to submit records to FASAMS in real time, eliminating the need for a monthly batch process for the MEs to submit records to FASAMS and improving the timeliness of DCF's data gathering.

Powered by Big Data

Big Data combined with AI and machine learning can be used for better insights and recommendations in areas such as client outcomes and expense estimation. For example, we can leverage historical data from past treatments, data captured by FASAMS and a new mobile app to deliver customized recommendations that can help provide a better CX experience while reducing operational costs.

3. Document Structure

This document below covers the following areas:

- **Project Approach** - This section discusses the approach the OZ team took for this project.
- **FASAMS System Overview / Background and overview** - This section contains background information about why and how the system came to be developed.
- **Critical Remediation Activities –** To get the system to work, we believe that these activities need to be performed prior to any further enhancements or modifications to the FASAMS system.
- **Detailed Recommendations** - Recommendations are found throughout the document, at various levels of the system, from the high/macro level to the specific/micro level. In this section we have compiled all recommendations from the review document.
- **Relevant quotes** – relevant quotes from interviews with DCF staff, the provider and MEs are included in this section.
- **User Experience Review** - This section includes summaries of interviews with various DCF employees, managing entities and a provider. It will discuss the modules that are part of the FASAMS system: Submission, Reports, Administration and Configuration. In each section, challenges with the system and recommendations are presented. A section on what we heard about the data accessibility is included. Finally, this section contains information about how the OZ team attempted to trace the QA test cases back to the original requirements.
- **Technical Review** – this section will include findings about known issues from a technical perspective; the tools and technologies that were used to build the system, the database, security, hardware and reporting tools. Recommendations for improvement are also included.
- **Appendices** – Supporting documents and information.

The intention is that each section of this document can stand on its own in case it needs to be reviewed independently so there is some duplication of information from section to section.

4. Project Approach

The OZ team went to DCF offices in Tallahassee for two days (April 24-25, 2019) and met with various stakeholders that use FASAMS or the data extracted from it. The OZ team also spoke with FEI Systems (FEI), the vendor who was contracted to develop FASAMS. In the following weeks, the OZ team conducted additional interviews with a provider and a number of Managing Entities.

The consistent message from people who are using the system is that:

- It does not match the reports provided by the MEs
- Is not easy to use and takes more time than the old system
- The data provided in the reports is not accurate
- It is difficult to extract data from the database for reporting

The OZ team reviewed the SharePoint site Libraries provided by DCF and FEI, which included requirements documents, training materials, test cases, diagrams, source code (for only SSRS and SSAS – we did not have access to the main application source code), test files and other documentation. In addition, the OZ team accessed and used the application and ran reports.

The OZ team did not receive access to all the source code. The review of that source code revealed some components were not included in the files sent to the OZ team.

The OZ team performed an initial review and submitted an initial assessment document (on May 9th with findings based on the initial conversations with DCF, and our review of the system to that date).

On May 9th, the OZ team provided the document below to DCF with initial findings. The team then continued to conduct interviews with MEs and further review the application and documentation.

In that initial document, we stated that "overall, based upon the kickoff sessions, the OZ team is not convinced that the system, as it is currently, supports the overall purpose for the system, and is highly likely to be non-functional for that purpose."

Subsequent to the initial review and assessment, the OZ team reviewed the documentation provided by DCF and FEI, which includes the requirements documents, training materials, test cases, and source code (SSRS, SSAS), and test files. In addition, the OZ team used the application to submit test data and ran all the reports (multiple times). We experienced "time-outs" and long wait times during the submissions process and the report generation process, and also submissions that did not have any status changes from the initial "Initialized" status. We viewed inaccuracies in the FASAMS reports. We performed an architectural review, analyzed the technology stack used for the application, reviewed the database: including the structure, schemas, tables, size, indexes, volumes, stored procedures, and the report definitions (RDL).

The OZ team attempted to trace the QA testing to the original requirements and was not able to map the test cases back to the original requirements. We spoke with the DCF resources who performed QA on the system and were told that the test cases were created directly from the Functional Design Document as they were not provided a copy of the requirements document or Amendment 002. We then attempted to map a sample of 30 requirements from Amendment 002 to the FASAMS Functional Design Document. Of the 30 items, 16 were found and 14 were not. As a result, we were not able to locate the QA test cases for all 30 items.

Based on our review and discussions, we found that the reports are running more quickly, but other reports are timing out or taking an inordinate amount of time to run. The other items still have not been resolved. Submitting files to FASAMS takes much longer than the old system and the MEs spend all month submitting files and correcting errors. Users cannot tell what records made it into the system. Some reports display incorrect results. While there are some standard reports available, users cannot access all the data that they need, they are not able to filter reports in the way that makes sense to them, they cannot slice and dice data in reports easily, and they cannot download a report that they can upload into their system for reconciliation.

INITIAL HIGH-LEVEL FINDINGS DOCUMENT:

Initial High-Level Findings

Overview (based upon our current understanding from the information shared during Kickoff meetings)
• At its core, the FASAMS System was developed to answer one (five-part) question:
  ○ Which Clients?
  ○ Received what services?
  ○ From whom
  ○ At What cost
  ○ For What Outcomes?
• It was delivered for production on January 1st, 2019.
• The original requirements gathering was started three years prior (in 2016) to delivery. It took a good part of a year to complete and was completed by North Highland.
• The procurement process to select the vendor for development of FASAMS system took place the following year (although did not take the entire year of 2017).
• The main development effort for FASAMS system took place during 2018.

Initial High-Level Findings
• During the development, FEI the organization chosen for development, confirmed the business requirements for the system through several JAD (Joint Application Development) sessions. • The system was to be delivered with both an OLTP / Transactional database, and a data warehouse specifically for reporting, which is the main purpose of the system. • Providers do not enter their information directly into FASAMS. Their activity information is entered into their Managing Entity's (ME) system. The ME then consolidates their providers' data and creates an XML file that is then uploaded into FASAMS (there are 7 different backend ME systems, although some (2-3) managing entities use the same platform) • Data should be uploaded to FASAMS by the 18th of the month for the previous month by the MEs • Between 25 and 30 reports can be generated from within FASAMS • FEI is hosting FASAMS on their AWS cloud. Their environment was certified by DCF's Amazon AWS team.
Issues (based upon our current understanding from the information shared during Kickoff meetings) • Reports generate slowly, and some will 'time out'. Although the OZ team was told that some reports are running quicker, there were reports that did time out during our demo. • Some reports were said to be 'cached' for performance because they were running slow in real time. • Reports are run against the production database, and not the data warehouse. Although the team was told that some reports are run against the data warehouse, the exact number that are run against each database is unclear. • The reports seem to be inaccurate, at the field level. There are some fields that have different usages depending on conditional data in other fields, there were instances during the demo where the logic of the reports seemed to be ignoring the conditional data. For example, during the initial demo, we saw one instance where a line item from a Provider had an associated cost of over $975,000,000, which seems to be an error. • It is unclear whether the issues in the reports are from the MEs data upload process, the business logic of storing the data in the OLTP database, or in the logic of generating the reports, or a combination of the 3. • The reporting module seems to generate static reports, without the ability to either select multiple entities to begin with, or to 'drill down' on the data. And while the report data set can be exported to a spreadsheet, the resulting spreadsheet is tabular and not in a form for true data analysis without knowledgeable "Excel" data analysis skills. • There is some doubt from members of the larger FASAMS team that we spoke to about whether the FASAMS system in its current state is supporting not only the business rules that were originally set, but also new rules that have come into effect since then. The example of "Co-Occurring" conditions was used -where there was no 'guidance' from the system about how to upload such a treatment incident. • MEs get a report of which records in their upload files had errors, but they do not have access to FASAMS (or a report) to see which records from their files were successfully uploaded to the system during upload, or an ability to verify the data after upload. • Providers do not have access to the FASAMS system either for upload of information directly or for reporting. • The OZ team was told the FASAMS OLTP production database is fully normalized, which seems to be creating performance issues. • When the Data and Reporting team needs to create a report / data query from the system they are accessing the OLTP production database directly, and the SQL Statements they create have many table 'joins' even for a simple query. • The OZ team was told there is no de-normalized/reporting data base layer. • The OZ team was told there was a data warehouse delivered with the system, yet it does not seem to be used for Ad-Hoc reporting or many reports generated within the FASAMS system • There are usability issues that were noticeable during the initial demo, and that were mentioned during our kickoff meetings and sessions. Some examples:

Initial High-Level Findings
○ "Dropdown" fields that are much shorter than the actual selection items in the dropdown with the first characters of numerous items being the same, which makes it difficult for users to select the correct item. ○ Headers on reports that are lost as the user scrolls or pages through a report, the placement of items. ○ Inability to select multiple items for a report when that would be a natural use case for users. Overall, based upon the kickoff sessions, OZ is not convinced that the system, as it is currently, supports the overall purpose for the system, and is highly likely to be non-functional for that purpose.
Initial Recommendations In addition to the User Experience and Technical Assessment originally scoped • Conduct a Business requirements mapping to the delivered system • Conduct an assessment of the QA/testing performed prior to deployment • Assess the implementation of SSRS as the reporting platform/module • Potentially, conduct a full Quality Assurance activity on the FASAMS system • Field by field evaluation of all reports for accuracy

5. System Overview/Background

The 2015 Florida Legislature, through Specific Appropriation 302A, requires the Department of Children and Families (DCF) to develop and implement a uniform information management and fiscal accounting system for providers of community substance abuse and mental health services. In response to this legislation, the Substance Abuse and Mental Health (SAMH) Program Office, in collaboration with the department's Office of Information Technology, created the Financial and Services Accountability Management System (FASAMS) project. This new system was completed and deployed on 12/31/2018.

The Florida Department of Children and Families (DCF) has requested that the OZ team perform a User Experience (UX) and Technical review for the FASAMS application.

Specifically, DCF wanted a series of assessments to be completed on the new system to help identify where improvements might be needed. This would include:

- A review and assessment of the application's operation and user interface design
- A review and assessment of the user experience for each role in the system
- A technical review and assessment of the design and implementation of the data warehouse and ETL processes
- A technical review and assessment of the design and implementation of the system reports
- A technical review and assessment of the overall performance of the system
- A review and assessment of the delivered system documentation.

According to DCF's Pamphlet 155-2, the Financial and Services Accountability Management System (FASAMS) was developed to collect and report "data on persons served not only in state-contracted community substance abuse and mental health provider agencies, but also in state contracted or state-operated mental health treatment facilities. Persons receiving state-contracted services include individuals who meet the priority population criteria for mental health or substance abuse. This data is needed, at the federal, state and local levels, to answer the management question 'who received what services from whom at what cost to achieve what outcomes'".

FASAMS was created to replace the existing Substance Abuse and Mental Health Information System (SAMH) and meet the following statutory requirements:

- A uniform management information and fiscal accounting system for use by providers of community substance abuse and mental health services;
- A uniform reporting system with uniform definitions and reporting categories;
- An integrated system with automated interfaces to Florida Medicaid Management Information System (FMMIS) and Child Welfare (FSFN) systems.

Our understanding is that the FASAMS System was developed to answer one (five-part) question:

- Which clients?
- Received what services?
- From whom?
- At what cost?
- For what outcomes?

The system was delivered for production on January 1st, 2019.

- The original requirements gathering process started in 2016 and was completed by North Highland.
- The procurement process to select the vendor to develop FASAMS system took place during the following year, 2017.
- FEI was selected to develop the application.
- The main development effort for FASAMS system took place during 2018.
- During the development, FEI confirmed the business requirements for the system through a number of Joint Application Development (JAD) sessions.
- The system was to be delivered with both an OLTP / Transactional database and a data warehouse specifically for reporting.
- Providers do not enter their information directly into FASAMS. Their activity information is entered into their Managing Entity's (ME) system. The ME then consolidates their providers' data and creates XML files that are then uploaded into FASAMS (there are seven different backend ME systems, although a few MEs use the same software vendor's application.
- Data must be uploaded to FASAMS by the 18th of the month for the previous month by the MEs.
- Between 25 and 30 reports can be generated from within FASAMS.
- FEI is hosting FASAMS on their AWS cloud. Their environment was certified by DCF's Amazon AWS team.

6. Critical FASAMS Remediation Activities

To get the system to work, we believe that the following critical remediation activities need to be performed prior to any further enhancements or modifications to the FASAMS system:

1. **Perform a full Quality Assurance (QA) test of the system**. Compare the delivered system to the final requirements document (LU915 - Amendment 0002 - Executed – Redacted.pdf). All types of records should be added, modified, and deleted. Each step of the import process should be tested - from initial upload to staging tables, to production, to data warehouse tables, to reports; including validation of min/max values, data types, required fields and dependencies. Once QA is completed, and the results documented, solicit feedback from internal and external users for a user acceptance test – this will help to identify items that meet the requirements and are technically correct but do not support business rules. The output of this process will be a prioritized list of items that need to be remediated. This should be the next step the DCF FASAMS team performs, prior to any further enhancements or modifications to the system. Without

understanding where errors are being introduced into the system, any further modification may be propagating those errors, and in the end making it harder to identify where fixes should be applied.

2. **Overhaul the database and architecture.** There are several ways that the database can be optimized to improve performance and the fix the foundation. This will enable the system to grow and support application fixes/enhancements, and will also improve speed, the user's experience, and data accuracy. The OZ team recommends making the following changes: convert the existing snowflake schema to a star schema; add/fix/remove database indexes; partition tables/indexes; and ensure that the production and dev/test databases are on different environments. Additionally, the staging database tables, the production database tables, and the data warehouse tables are all implemented within the same database. These should be partitioned out to separate databases. These changes are needed to help fix the submission process and improve performance (see details in Technical Review Section).
3. **Fix the submission process.** The OZ team identified many changes needed to the submission process, but the most critical ones are: speed up the import process so that it does not take hours (or overnight) to import ME submissions; allow users to get a clear picture of the import status (what file is being processed, what record is being processed, and what percentage is completed); make it easy for submitters to see what the errors are and to how fix them; create a reconciliation process so that submitters will be able to tell what records were successfully imported into the system and which ones failed; and allow users to cancel or roll back an import file that they may have submitted by mistake or wish to cancel / rollback for other reasons.
4. **Create a new, separate reporting database (data warehouse).** This will provide multiple benefits: it will improve the time it takes to run reports; remove processing burden from the transactional system and database; and the DCF Data and Reporting Team will be able to create new reports and ad hoc reports quickly and easily.
5. **Fix the existing reports.** Determine the cause and correct the reports and/or underlying data so that the reports render with correct data; create a way for users to export reports so that they can use the export files for analytics or import into other systems; add additional prompts for filtering/grouping/sorting the report results; and update the way that reports are displayed on the screen. This might involve adding new reporting toolsets, such as power BI as an example.
6. **Fix the overall user experience of FASAMS**, from providing feedback to users about the processes as they are running, to selection criteria of the reports.

7. Detailed Recommendations

Below is a list of issues that were either reported to the OZ team by users of the system or that we discovered while using the system. Our recommendation(s) are provided along with each issue. Please note that this list contains a short description of the issues and recommendations. Please click on an issue description and the link will take you to the section of the document that contains more details about the issue and the recommendation(s) on how to resolve the issue.

1. MEs spend a lot of time fixing errors
    - Recommendation: Provide an easy way for the user to identity the record and field that caused an error.
    - Recommendation: Group multiple errors on a record together to make it easier to fix them all at once.
2. Process takes a long time
    - Recommendation: The system should be updated so that files are processed quickly, and submitters get feedback the same day.

3. Large files take a long time
    - Recommendation: The system should be re-architected so that importing a 50MB file takes no longer than one hour.
4. Users want to see the files that they submitted
    - Recommendation: Add the original file that was uploaded with the submission results so that the original file can be viewed.
5. It's easy to submit files in the wrong order by mistake
    - Recommendation: Create a way for the user to identify the dataset type for a file when it is uploaded for import. The system should then be smart enough to process the files in the correct order.
6. Users want a reconciliation process
    - Recommendation: Provide a reconciliation report for each file submitted the file
        - Total number of records successfully imported
        - Total number of records that were rejected
        - Total number of records that contained warnings
        - Detail of records successfully imported
        - Detail of records that were rejected
        - Detail of records that contained warnings
    - Recommendation: When viewing a job in submissions the system should display the following:
        - Total number of records in the file (new column). This is not clickable.
        - Total number of records successfully imported (existing column – make it clickable). When the user clicks on the number the system will display the records that were successfully imported. This needs to be downloadable in Excel.
        - Total number of records that were rejected (existing column – make it clickable). When the user clicks on the number the system will display the rejected records. This needs to be downloadable in Excel.
        - Total number of records that contained warnings (existing column – make it clickable). When the user clicks on the number the system will display the records that contain warnings. This needs to be downloadable to Excel.
7. Users want to be able to remove files submitted by mistake or with errors
    - Recommendation: Allow users to back out a file that was imported into the system – exploring the impact and the specific business rules would need to be defined to specify what files are eligible to be removed and which ones are not.
8. Users would like web services to be easier to use
    - Recommendation: Allow users to connect with Web Services/ API and add the appropriate security to the web service so that the server does not need to connect to the VPN.
9. Easy to upload to upload to the wrong customer
    - Recommendation: Provide a confirmation screen with the summary of what the user are about to upload and for which ME.
    - Recommendation: Allow users to back out a file that was submitted by mistake.
10. It's hard to tell what is going on
    - Recommendation: Provide runtime feedback when a file is submitted so that the user can tell if the file is being imported, exactly where it is in the process, or if there is an error. In addition, provide a way for the user to cancel a job.
11. System does not give immediate feedback when there is an error
    - Recommendation: Provide immediate feedback when a file contains an error.
    - Recommendation: Provide a way for the user to cancel a job.
12. No way to filter or group jobs on submission screen
    - Recommendation: Consider adding a filtering functionality that will provide the ability for users to view by various Datasets, as needed as well as a function to group that data.
13. No way to cancel once user clicks Upload button

- Recommendation: Light dismiss (clicking on another area on the screen) should be used on this screen to allow the user to switch to a different task without friction OR add an 'X' icon or other close function on this window.

14. No way to remove files from upload once selected
    - Recommendation: Add the ability to remove files from the list.
15. Reports are incorrect
    - Recommendation: Perform a full Quality Assurance Review for each report to ensure that what is appearing on the report matches the requirements. Requirements may also need to be reviewed to confirm that they make sense. Once the review is complete, identify the cause and make the appropriate corrections.
16. Custom reports are not available
    - Recommendation: Provide the ability to create a custom report.
        - User should be able to select at a field level
        - User should be able to save the custom report in the end user profile
        - Ability to share custom created reports with other end users (admin level)
17. Additional prompts are needed
    - Recommendation: Evaluate what additional prompts are needed on reports and add them.
18. Report export files are difficult to use
    - Recommendation: Add the ability to save the data and the column headings only. When the user chooses to export the data, provide a version of the report that contains only the headings and actual data (no page numbers, repeated headings, etc.). Formatting should be stripped (except for dates, phones numbers, etc.).
19. Headings need improvement
    - Recommendation: Update report descriptions, prompts and headings to use correct business language.
20. Report viewing area too small
    - Recommendation: Consider relocating the report 'Categories' selection functionality located in the left-hand pane to another area in order to utilize the full width of the page to view the report after it is run.
21. Report names are not consistently shown on the report results view
    - Recommendation: As standard practice the name of the report should be listed at the top of the page so that it is clear what report the user is running. Highlighting the report name on the left navigation menu will help as well.
22. Drop down values do not display completely
    - Recommendation: The width of the field drop down window should be widened so that the end user can read the value completely without having to use horizontal scroll bar on the bottom.
    - Recommendation: The length of field and drop-down window should be lengthened so that the end user can read the value completely without having to scroll from the top to the bottom.
23. Hard to find values in dropdowns
    - Recommendation: Employ a searchable drop-down list in reports.
24. When scrolling down a report, column headers do not remain in view
    - Recommendation: Keep column headers visible when scrolling through data on any given report.
25. Username and email cannot be changed
    - Recommendation: Add a function to rename a user account and change email address or to link/migrate data to the new user account.
26. The administration page looks very busy and can be difficult to read
    - Recommendation: Create sub-navigation under the 'Administration' section of the website for each of the areas (Submitting Entities, Messages, Users, Groups and Roles) that will take the user to a page to display each of the areas in the full viewing area of the page.
27. Additional functionality for each area is hidden behind menus

- Recommendation: Employing the above recommendation would provide enough space to leave additional options to be listed directly on canvas.

28. Most sections require scrolling as well as pagination to view the lists within each section
    - Recommendation: Using the full page will increase viewing area and illuminate the need for scroll bars as well as provide opportunities to add more functionality on canvas to expose all options available to the end user.
29. Too many sections as well as content on the configuration page make this page somewhat difficult to read
    - Recommendation: Add sub menus to the primary navigation for Configuration that will take the end user directly to the respective page for Vocabulary Versions, Rules and Dynamic Data Sets. This will provide a larger viewing area for the content as well as reduce scrolling and pagination in comparison to how it is currently presented.
30. Additional functionality for Dynamic Data Sets are hidden behind the menus
    - Recommendation: Employing the above recommendation would provide enough space to leave additional options to be listed directly on canvas and eliminate the need for an ellipse.
31. It is not obvious that the 'Name' values in each row are clickable
    - Recommendation: Add a color change when the end user hovers over the text to make it obvious that it is a link.
32. Database performance issues/correct design deficiency
    - Recommendation: Partition tables and indexes
    - Recommendation: Optimize indexes
    - Recommendation: Convert snowflake schema to star schema
    - Recommendation: Create a separate reporting database
    - Recommendation: Hardware improvements
    - Recommendation: Lower cardinality
    - Recommendation: Minimize the number of columns
    - Recommendation: Use less expensive data types
33. Difficult to create ad hoc and new reports
    - Implement Power BI for reporting
34. Users need to be able to mark records as “deleted”
    - Recommendation: Add a function to the application that allows users to mark a record as deleted.
35. Data becomes orphaned
    - Recommendation: Allow users to delete records and do not allow orphaned records.
    - Recommendation: Add constraints on the database that will not allow orphaned records.
36. Data does not match provider records
    - Recommendation: Perform full QA check of the system.
37. Data does not match MEs manual records
    - Recommendation: Perform full QA check of the system.
38. Data is inconsistent between providers
    - Recommendation: Create a process that ensures consistency between providers and MEs.

8. Relevant Quotes

During the course of this review, the OZ team interviewed a number of stakeholders. We consistently heard common themes and statements about FASAMS, and a number of quotes from those interviews are used as support for the various sections throughout the document. We have compiled some relevant quotes from these interviews (and the document below) that we feel illustrate these main themes.

- *“I THINK THAT THE INITIAL UPLOAD IS THE EASY PART. THE FIXING OF ERRORS IS A CHALLENGING ONE AND TIME CONSUMING.”*
- *“YOU KNOW, HISTORICALLY AND IN THE PREVIOUS SYSTEM, IF THERE WAS AN ERROR, I COULD SEE THE EXACT RECORDS FROM THE SYSTEM THAT WAS ERRORED. SO, I QUICKLY BE ABLE TO GO EXACTLY TO THAT FIELD THAT HAD AN ERROR. IF THERE WERE MULTIPLE ERRORS, THEY WERE GROUPED TOGETHER, THAT PARTICULAR RECORD AND YOU COULD MAKE THOSE FIXES AND MOVE FORWARD.”*

- *"RIGHT NOW, WITH THE LIMITATIONS OF THE SYSTEM, WE'RE SPENDING BASICALLY TWO DAYS UPLOADING AND THE REST OF THE MONTH TRYING TO GO THROUGH THE ERRORS THAT'S, RETURNED TRYING TO CLEAR THESE UP AND DRILL DOWN TO WHAT THE REAL ISSUES ARE."*
- *"IT'S A VERY LENGTHY PROCESS TO GENERATE THOSE RECORDS AND THEN OF COURSE, TO UPLOAD THEM, ESPECIALLY THE CLOSER WE GET TO THE DEADLINE FOR DATA SUBMISSION EVERY MONTH, IT SOMETIMES COULD TAKE UP TO A DAY FOR US TO HAVE ALL OF OUR RECORDS PROCESSED. SO, FOR THAT 24-HOUR PERIOD, WE DON'T KNOW REALLY WHETHER OR NOT THERE'S ANY ISSUES WITH THE DATA OR WHETHER OR NOT THERE'S GOING TO BE ANY REJECTIONS."*
- *"WE ENCOUNTER UNEXPECTED ERRORS FREQUENTLY. AND WITH THOSE UNEXPECTED ERRORS, WE HAVE NO WAY OF KNOWING WHAT DATA GOT IN BECAUSE WE DON'T HAVE ANY ACCESS TO SEE THE DATA THAT ACTUALLY MAKES IT INTO FASAMS".*
- *"WE WENT LIVE, BUT I'M NOT SURE WE WERE READY. RIGHT. BECAUSE THERE'S ALL THIS VALIDATION THAT I DON'T THINK GOT DONE AT A REAL END USER LEVEL YET."*
- *"I DOWNLOADED 10,000 DEMOGRAPHIC RECORDS FOR <GEOGRAPHY>. I PICKED THE WRONG ME AND SUBMITTED ON THE BBCBC, ALL OF A SUDDEN, I HAVE 10,000 RECORDS ON BBCBC THAT I CANNOT FIX BECAUSE THEY DON'T BELONG TO THEM JUST BECAUSE IT'S A CONTRACT WITH A CONTRACT IDS THAT DON'T BELONG TO THEM. BUT IT'S STILL HOLDING THEM ON THE OTHER SIDE, SO THERE IS NO WAY THAT IT WAS NO WAY TO REVERT THIS RECORD BACK. I CANNOT DELETE THEM BECAUSE THEY DON'T EXIST. I CANNOT FIX THEM BECAUSE THEY DON'T BELONG TO THAT ME. SO HOW DO WE TAKE THEM BACK? IT STILL COUNTS AS AN ERROR ON OUR SIDE."*
- *"WE LOST CONNECTIVITY AGAIN."*
- *"SO, THE BIGGEST BARRIER RIGHT NOW, THE LARGEST FACTOR, IS THE AMOUNT OF TIME IT TAKES TO GO THROUGH THOSE ERRORS AND ALSO NOT BEING ABLE TO PULL BACK OUT WHAT'S IN THE SYSTEM. SO, WE'RE KIND OF OPERATING BLIND WHERE WE SUBMIT SOMETHING. WE WOULD RESUBMIT EVERYTHING BECAUSE IT'S EXTREMELY CUMBERSOME TO FIGURE OUT WHICH PARTICULAR RECORD FAILED AND THAT'S KIND OF WHAT I'M GETTING AT. SUBMITTED TEN THOUSAND RECORDS AND THERE'S A THOUSAND ERRORS, WE TRY TO FIX AS MANY ERRORS AS POSSIBLE. JUST RESUBMIT IT AND SEE WHAT WE'RE LEFT WITH AND TRY TO NARROW IT DOWN TO WORK WITH IT."*
- *"THERE'S A LOT OF GUESSWORK THAT'S GOING INTO THIS WHOLE PROCESS RIGHT NOW."*
- *"WE HAVE TO REPEAT THAT WHOLE PROCESS SOMETIMES TWO, THREE, FOUR TIMES WITHIN A GIVEN MONTH."*
- *"OUR WORKLOAD HAS LITERALLY DOUBLED UNDER FASAMS THAN IT DID HISTORICALLY UNDER, UNDER SAMH."*
- *"THE SUBMISSION PROCESS, OUR PROCESS, STARTS ON THE FOURTH OR FIFTH OF THE MONTH. OKAY? AND IT TAKES US BETWEEN THE FOURTH THROUGH THE 18TH, IN ORDER TO GET EVERYTHING SORTED, VALIDATED PROCESSED AND THEN UPLOADED INTO FASAMS. AND THEN ONCE THAT IS DONE, THE REST OF THE MONTH IS BASICALLY JUST TROUBLESHOOTING, ERROR CORRECTING AND REPROCESSING DATA. SO, IT'S A CYCLE THAT DOES NOT END. IT'S LITERALLY THE ENTIRE MONTH, ALL MONTH, 40 HOURS A WEEK, NONSTOP."*
- *"THE MES HAVE BEEN ASKING US SINCE JANUARY TO GET ACCESS TO THE DATA SYSTEMS SO THEY CAN SEE WHAT GOT IN THE SYSTEM. SO, IF THEY SEND A HUNDRED RECORDS TO US AND WE SAY 10 OF THOSE RECORDS FAILS, THEY GO AND THEY FIXED THE 10 RECORDS, BUT THEN THEY COME BACK TO US AND THEY SAY, WELL, I WANT TO SEE THE 90 RECORDS THAT GOT IN. AND IT'S LIKE, WHOA, IT'S THE ONES THAT DIDN'T FAIL."*
- *"ON ANY GIVEN DAY OF THE WEEK, MY TEAM IS LIKE, THE DISCHARGE REASONS DON'T SEEM TO MAKE ANY SENSE. OR THE LEGAL STATUS DOESN'T SEEM TO MAKE ANY SENSE WITH THE…, THIS DOESN'T SEEM TO MAKE ANY SENSE. SO, WE'RE PUTTING OUT FIRES ALL OVER THE PLACE".*
- *"MY GOAL IS TO GET THE DATA SYSTEM AND ITS FINANCIAL COMPONENTS TO REPORT AS ACCURATELY AS HUMANLY POSSIBLE WHAT IS ACTUALLY HAPPENING WITH THE SUBCONTRACTS. IT'S PRETTY FAR AWAY FROM THAT RIGHT NOW."*
- *"I CAN TELL YOU ON THIS REPORT, ALL THE VALUES OVER HERE ARE WRONG… THERE'S NO WAY THEY'RE PERFORMING AT 13.87 [PERCENT] BECAUSE IF THAT WAS THE CASE, THEY WOULD HAVE FIRED EVERYBODY."*
- *"THIS REPORT IS A PERFECT EXAMPLE OF A DISCONNECT BETWEEN THE BUSINESS AND THE SYSTEM."*
- *"I WANT THEM TO ALSO BE ABLE TO BUILD THE REPORTS THEY NEED WHEN THEY NEED THEM."*
- *"WE KNOW THAT THE END-USER REPORTS ARE STILL IN FINE TUNING PHASES."*
- *"THE ONLY CHALLENGE WE HAVE IS WHEN YOU GO TO THE ERROR REPORTS ON THE REPORTS, YOU KNOW, THEY CREATED A FUN, EASY WAY TO SHOW MAYBE TO A BUSINESSPERSON, BIG GROUPING OF THE CALCULATION BY ME, BY CHAPTER, ETC. BUT WHEN YOU TALKING ABOUT DAY TO DAY PERSON WAS DOING OPERATIONS AND TRYING TO FIGURE THOSE ERRORS, YOU KNOW, YOU GOT TO REMOVE THAT GROUPING THING, REMOVE THE EMPTY COLUMN, CLEAN A BUNCH OF STUFF JUST TO BE ABLE TO WORK ON THE FILE THAT DOESN'T HAVE ANYTHING ELSE BUT THE DATA OF WHAT RECORD HAS BEEN AFFECTED AND WHAT ERROR CODE SO WE WORK ON THE RESOLUTION OF THE ERROR ITSELF. THAT PROCESS ON OUR SIDE IS TAKING EASILY BETWEEN 30-45 MINUTES. JUST TO GET TO WHAT WE NEED TO DO TO START FIXING THE PROBLEMS. AND EVERY TIME IT'S TAKING MORE TIME BECAUSE THERE IS MORE DATA."*

9. FASAMS User Experience Review

As part of the user experience review, the OZ team conducted interviews with DCF team members, MEs and a provider. We include summaries of those interviews in the interview section. The team also used the application from the submission process to reporting – using the documentation as a guideline on how to use the system. We discuss the modules that are part of the FASAMS system: Submission, Reports, Administration and Configuration. In each section, challenges with the system and recommendations are presented. A section on what we heard about

the data accessibility is included. Finally, this section contains information about how the OZ team attempted to trace the QA test cases back to the original requirements.

Interview summaries
This section contains summaries of the interviews that the OZ team conducted in Tallahassee with various DCF departments, FEI, a provider and several MEs.

Submission module
The FASAMS system is comprised of a user interface that allows the Managing Entities (MEs) to submit files that contain information about providers, clients and treatment episodes to DCF on a monthly basis. These files are then imported into the FASAMS system so that DCF may answer the five-part question - who received what services from what provider at what cost to achieve what outcomes? The system is supposed to provide feedback to the MEs so that they know which records were successfully imported and which ones failed. The MEs then have 60 days to correct the failed records and resubmit them to FASAMS.

Reports module
Once the submission process for the month is completed, DCF employees will run standard system reports from the FASAMS system to answer the five questions. These reports are available in the reporting module. MEs have access to four reports in the reporting module.

Administration module
FASAMS also contains an administration module where users, submitting entities, messages, groups and roles are added and maintained.

Configuration module
The configuration module allows a user to add the vocabulary mappings, rules and dynamic data sets.

Data
This section contains summaries about what we heard from stakeholders about the challenges with the data.

Requirements and Test Cases
This section contains a description of the process followed by the OZ team attempting to trace the original requirements to the test cases and test results.

9.1.Interview Summaries

The following section provides a summary of information that was shared during interviews with different actors of the FASAMS system. The interviews summaries describe the experiences and opportunities for improvement shared by DCF's personnel and external DCF users (Managing entities and Providers).

9.1.1.FASAMS Demo

The demo was provided to the OZ team in person by the DCF Data and Reporting team. The Data and Reporting team are mainly tasked with creation of new reports that fulfill requests from other departments. In addition, the team maintains the integrity of the data.

Submission screen comments:

- On the Submission page, the column header disappears, and they are not labeled clearly enough

> *"As soon as I scroll off the headers, I lose any sort of indication of what this thing's actually, what this means. I mean, I presume this is submission date, but...I don't even know if it's submission date could be submission to it could be end date."*

- The Excel spreadsheet does not give me access to the data in a format-free way

> *"PDF isn't actually functional, meaning it's not useful. You can't do anything with it. Excel is formatted like a table so; it is difficult to do things with because you almost have to un-table it to be able to do any analysis stuff. If in the records that this managing entities submitted, there were nine errors, but let's say that they had a series of records that all have the same exact errors that were inside of those. They would have to manipulate the spreadsheet just to be able to group the errors together."*

- The system does not list what records passed

> *"The MEs have been asking us since January to get access to the data systems so they can see what got in the system. So, if they send a hundred records to us and we say 10 of those records fails, they go and they fixed the 10 records, but then they come back to us and they say, well, I want to see the 90 records that got in. And it's like, whoa, it's the ones that didn't fail."*

Reports comments:

- *"On any given day of the week, my team is like, the discharge reasons don't seem to make any sense. Or the legal status doesn't seem to make any sense with the..., this doesn't seem to make any sense. So, we're putting out fires all over the place".*
- Cumulative performance report – for reference, the target is 85%

> *"I can tell you on this report, all the values over here are wrong... there's no way they're performing at 13.87 [percent] because if that was the case, they would have fired everybody."*

- Disconnect between the business and the system

> *"This report is a perfect example of a disconnect between the business and the system. The top portion of this report is entitled Non-Subcontracted Services. So, you have providers and we generally work with a population that's indigent. You have providers that are providing services to individuals and they may not be compensated for that service. There are 15 pages of what the system considers to be uncompensated care. If that was the case they would be out of business because two thirds of the information that they put into the system, nobody's paying them for it. It's not that I'm not getting paid. There's a disconnect between the service events and on the subcontract not talking."*

9.1.2. Interview with DCF's Design and Policy Team

The Design and Policy team shared a detailed description of the conception of the FASAMS system, the factors that were considered, and events that occurred during implementation. This session provided some insight of the purpose of the FASAMS system, the target users of the system and the way data is collected through its physical representation (DataSets).

The FASAMS System was developed to answer one (five-part) question:

- Which clients?
- Received what services?
- From whom?
- At what cost?
- For what outcomes?

9.1.3. Interview with DCF's SAMH OITS Department

SAMH OITS team was not part of the design of FASAMS system. This team was introduced to FASAMS for incident management and account creations. SAMH OITS team recognizes that FASAMS included some improvements like:

- Date upload process in FASAMS has a data validation that was not present in SAMH

- The communication with users has improved compared to process before FASAMS

9.1.4. Interview with DCF's Data and Reporting Team

The Data and Reporting team are mainly tasked with creation of new reports that fulfill requests from other departments. In addition, the team maintains the integrity of the data.

The current implementation does not contain elements in the data handling and storage side that would make easier to build new reports. The team pointed out the following during our interview:

- They believe that the current implementation lacks a Data Warehouse to support the report engine.
- The need for delete/soft-delete functionality was identified once the system was in production. The team is currently writing SQL statements to delete/soft-delete data as needed.
- The way the data is structured in a cascading relationship creates data inconsistency when a soft delete is executed. Some data becomes orphaned and this creates issues with performance and report accuracy.
- DCF team would like to have a method/functionality/feature unnormalized view that will provide a better and faster way to produce Ad-Hoc reports without hitting the production database directly.
- Currently, ad-hoc reports are built using SQL queries with multiple joins that may affect performance once more users have access to run reports.
- Data and Reporting team have created some workarounds to speed up the way reports are built but the team recognizes that FASAMS should have been better designed knowing that the outcome was to produce reports to different DCF's departments.
- Access through VPN presents some issues due to sporadic disconnections.

9.1.5. Interview with DCF's Financial Analysis Team

This team's purpose is to perform financial monitoring of the managing entities. In general terms the team does onsite monitoring of managing entities' financial records, policies and procedures. Since the financial analysis team has access to information produced by Providers that is delivered to Managing Entities, they can compare raw data against what is delivered by FASAMS reports. The financial analysis team has identified the following issues with FASAMS:

- Financial team does not have knowledge of the final requirements agreed upon for the FASAMS implementation.
- The data produced by FASAMS reports is not reliable and it does not match with the information provided by the Providers.
- There is a need to allow access to the database or the report engine to generate new reports that may be required by customers.
- FASAMS is not stable enough to be trusted.
- FASAMS has not provided the expected functionality.
- The team expected to be able to produce reports at client level (patients) to allow the team an easier and more efficient way to validate the data provided by MEs with the data from FASAMS.
- MEs do not have a clear understanding of what needs to be loaded.
- Validation/Processing of reports takes too long (It could be caused by interface with external data sources)
- There are variations in the way that services are handled. For example, Provider A uses code "123" and Provider B uses code "456" for the same service. It makes it impossible to determine that the services are the same.

9.1.6. Interview with DCF's HQ Infrastructure Team

DCF's Infrastructure Team is not responsible for and does not have access to the FASAMS's infrastructure because FEI owns the environment. The infrastructure team described the system as an application as a service hosted in a third-party AWS's account owned by the vendor (FEI).

The infrastructure team partnered with an Amazon team (specialized in AWS) to evaluate the architecture proposed by FEI and make sure that the data integrity is maintained, and the infrastructure was well designed. The infrastructure team designed and implemented the way FASAMS is accessed (Direct connect) by users using either Aventail (DELL) VPN client for non DCF members or CISCO's AnyConnect for DCF members.

9.1.7. Interview with DCF's Managing Entity Contracts Department

The contract department manages statewide projects, contracts and business operations for DCF. The contracts team shared the following highlights during the interview:

- *"WE KNOW THAT THE END-USER REPORTS ARE STILL IN FINE TUNING PHASES."*
- *"MY GOAL IS TO GET THE DATA SYSTEM AND ITS FINANCIAL COMPONENTS TO REPORT AS ACCURATELY AS HUMANLY POSSIBLE WHAT IS ACTUALLY HAPPENING WITH THE SUBCONTRACTS. IT'S PRETTY FAR AWAY FROM THAT RIGHT NOW."*
- Manual records are trusted

  "WE COMPARE ME'S MANUAL RECORDS AGAINST FASAMS REPORTS."

- There are no custom reports

  "FASAMS IS LACKING SOME CUSTOM REPORTS USEFUL FOR FINANCING."

- They want a way to build their own reports

  "IT FEELS TOO FAR AWAY, MAYBE EIGHT OR NINE OTHER REPORTS THAT WE HAVE BEEN COLLECTING MANUALLY, ALMOST ALWAYS IN SPREADSHEET FORMAT. SO, YOU CAN DO THINGS WITH THEM."

- *PROJECT CODES CREATE ISSUES*

  "MANAGING ENTITY "A" DOESN'T HAVE TO PAY FOR SERVICE "B" IN THE SAME WAY MANAGING ENTITIES "C" DOES. MANAGING ENTITY "A" DOESN'T HAVE TO PAY THAT PROVIDER "B" FOR SERVICE "C" IN THE SAME WAY IT PAYS PROVIDER "D" FOR THE SAME SERVICE". "MANAGING ENTITIES CAN USE MULTIPLE METHODS OF PAYMENT". "MANAGING ENTITIES DON'T REALIZE THE BEST WAY TO REPORT WHAT IT IS THEY'RE QUITE LEGALLY DOING. THEY'RE CALLING IT A CASE RATE WHEN IT'S NOT, YOU KNOW, AND THAT'S CAUSING THE ALGORITHMS TO FAIL CAUSE THE CASE"

- Validation of reports from FASAMS needs to be done

  "I'M GOING TO PULL EACH MANAGING ENTITIES REPORT, REACH OUT TO THE REPOSITORY FOR THE MANUAL PROVIDER DETAILED FINANCIAL REPORTS, MATCH THEM UP AND SEND IT OFF TO THAT CONTRACT MANAGER, GIVE THEM A WEEK TO IDENTIFY ANY VARIANCES FOR ME."

- *"REPORTS (THE EXPENDITURE DETAIL REPORT) IS NOT IN BUSINESS LANGUAGE! VOCABULARY IS CONFUSING. THE VOCABULARY USE FOR SOME OF THE UNITS OF MEASURE WAS MISLEADING. IT COULD BE SIMPLIFIED."*

- Some fixes should be applied to the reports like the one mentioned in this quote

  "RIGHT NOW, PULLING OR RUN THE EXPENDITURE REPORT, THE VERY FIRST SET OF DATA YOU SEE IS ALL THE DATA WE'RE NOT PAYING FOR. IT'S KIND OF POINTLESS, YOU KNOW, WHY IS THAT AT THE TOP?"

- *"I WANT THEM TO ALSO BE ABLE TO BUILD THE REPORTS THEY NEED WHEN THEY NEED THEM."*
- *"WE WENT LIVE, BUT I'M NOT SURE WE WERE READY. RIGHT. BECAUSE THERE'S ALL THIS VALIDATION THAT I DON'T THINK GOT DONE AT A REAL END USER LEVEL YET."*

9.1.8. Interview with FEI's Development and Architecture Team

The team that participated in this meeting works for the company that developed and implemented the FASAMS system. The meeting focused on FEI's team describing the process of development of the tool from the initial understanding up to deliverables.

The initial requirement shared with FEI's team was to develop a team to import a lot of data into a transactional system to then transform it into a tool for reporting on the migrated data. FEI understood that DCF had a kind of dual goal, they're giving users the ability to get all the data and reports on a daily basis versus having some more specialized reports that are more de-normalized and built to answer more specific questions.

9.1.9. Interview with a Provider

The person interviewed during this meeting works for one of the largest providers that is contracted with DCF to provide mental health services. The provider does not send data directly into FASAMS, instead they send the information to their ME through a third-party solution (Five Points) provided by the ME.

The provider identified some advantages that FASAMS offered over the old way of managing the data for the state funded services offered.

- FASAMS is replacing the manual work done over the last 20 years.
- The redundancy of the data was reduced therefore the amount of data in general was also reduced.
- The data uploaded to FASAMS allows better and more accurate reports. This also improves the spending process.

The Provider shared some knowledge of why providers send data to MEs instead of directly into FASAMS - it is because some MEs have invested a lot in creating the tools to manage the data from the provider that then is formatted and uploaded into FASAMS.

The experience shared by the provider is not related directly with FASAMS. The experience described confirms that when the provider sends data to the ME, the feedback offered by the tool used helps a lot in the moment of correct errors. The process follows the steps below:

- A set of comma delimited files are uploaded to the ME through the Five Points application.
- A Pre-FASAMS module receives the data and executes business rules.
- A report of the process is displayed where the provider can view the following:
    - Number of records uploaded
    - Records that failed which can be corrected in the source for re-upload or directly in the tool.
- The system (Five Points) delivers reports quickly.

The rules applied to validate the data in the Pre-FASAMS module are the same used in FASAMS and based on the interviewee's expertise in the mental health services field; some of the rules are cumbersome and don't make sense from a business standpoint. This validation affects the payment of service by DCF/MEs – because some services cannot be reported they are not paid for.

When providers are required to send the data for the services provided, some data is fixed in a way, so they can get paid. This creates data inconsistencies that affect FASAMS reliability.

Some of the challenges that FASAMS exposed to the providers are described in the below statement extracted from the conversation/interview:

- I think everybody in the state is going to be working through that treatment episode chapter, which on the face of it looks straightforward. It looks okay, whenever a patient moves between these seven or eight different levels of care, which encompass 45 to 50 different covered services by the state will come, we'll pay for, then you do admissions and discharges as a patient goes through the system. However, trying to manage that in an EMR system and trying to make sure that you have no white admission and the services associated with that admission done properly is extremely challenging.
- For decades, providers have been asking to see what they submitted so they can confirm the accuracy of the system. According to the provider,

    "I DON'T KNOW HOW MUCH OF MY DATA ACTUALLY MAKES IT ALL THE WAY. AND ONCE IT MAKES IT THERE, WHOEVER'S RUNNING THE OUTCOME REPORTS, I DON'T KNOW HOW THOSE ARE CONSTRUCTED."

9.1.10. Interview with Managing Entity (South East Florida Behavioral Health Network)

South East Florida Behavioral Health Network is one of the Managing Entities that send data to DCF throughout FASAMS. SEFBHN serves Indian River, Martin, Okeechobee, Palm Beach and St. Lucie counties. SEFBHN's experience with FASAMS has not been the ideal and here are some of their concerns:

- SEFBHN expressed that there is a disconnect between FASAMS being an I.T. project versus a project that actually drives "clinical decision making" and not necessarily in a good way.
- The upload data produces reports that affect the resources distribution because the reports do not represent the real accountability of services served. One of the reasons for this could be the use of "Optionally required fields" that some providers will mark the fields as not required and this generates data inconsistency.
- There is not an identified resource/person that is responsible for ensuring the data integrity.
- Once the upload process is completed, the generated report contains information about errors that occurred during process. The way the errors are described does not help much in the resolution process. A sample scenario will be when ME uploads 18 different records that go up and record #2 fails, #3 through #18 won't process because #2 wasn't accepted. But the error report will just indicate an error, not that record #17 failed or record #15 failed. There is no way to know what was accepted and what not.
- SEFBHN provided a separate tool that is used by some Providers to upload data. This tool generates reports that are used by SEFBHN instead of the ones produce by FASAMS. They do not trust the reports generated by FASAMS.
- SEFBHN raised the concern that most of the MEs (maybe all) do not understand the final requirements mandated by DCF. This is an issue because they may not be submitting what DCF is expecting.
- There needs to be a better notification and training process when DCF adds new features to the system.
- DCF has included new options to the system that SFEBHN does not fully understand and it could be because there was not an effective way of communicating the change.
- There is a disconnect between the MEs and DCF's FASAMS.

Second Interview:

- Access to the historical data is limited or nonexistent.
- The feature available to upload data through SFTP is not being used because users prefer an interface that provides feedback. A quote from the interview explains their concern:

  "SO, WE'D RATHER SEE WHAT WE'RE LOADING, AND WE'LL HAVE A BETTER CONTROL OF THAT FILE IF I WOULDN'T DO OR NOT".

- The feature available to upload data through Web Services is not being used because it requires VPN connection and this connection does not work in the background. It requires human interaction.

  "WE NEED TO HAVE THE WEB SERVICES CONNECTED ON THE VPN. SO, IF IT IS NOT A BACKEND PROCESS, I DON'T KNOW HOW WE COULD GET OUR SERVERS CONNECTED TO THE DCF"

- We work with two MEs today – if we submit to the wrong one we cannot fix it

  "I DOWNLOADED 10,000 DEMOGRAPHIC RECORDS FOR SOUTHEAST FLORIDA. I PICKED THE WRONG ME AND SUBMITTED ON THE BBCBC, ALL OF A SUDDEN, I HAVE 10,000 RECORDS ON BBCBC THAT I CANNOT FIX BECAUSE THEY DON'T BELONG TO THEM JUST BECAUSE IT'S A CONTRACT WITH A CONTRACT IDS THAT DON'T BELONG TO THEM. BUT IT'S STILL HOLDING THEM ON THE OTHER SIDE, SO THERE IS NO WAY THAT IT WAS NO WAY TO REVERT THIS RECORD BACK. I CANNOT DELETE THEM BECAUSE THEY DON'T EXIST. I CANNOT FIX THEM BECAUSE THEY DON'T BELONG TO THAT ME. SO HOW DO WE TAKE THEM BACK? IT STILL COUNTS AS AN ERROR ON OUR SIDE."

- Sample of misleading errors

  "SO, LET'S SAY FOR SOME REASON THE CONTRACT NUMBER WAS INCORRECT. FROM THERE, YOU GET ADMISSION IS INCORRECT. THERE IS NO PERFORMANCE OUTCOME MEASURE AND THERE IS NO DIAGNOSIS ON THE ADMISSION. SO FAR FIVE ERRORS,

WHICH ARE TRUE. BUT THE REASON WHY THEY ARE TRUE IS BECAUSE ONLY ERROR WITH THE CONTRACT. NOW THERE'S SOME CONTRACT NUMBER WAS INVALID AND OF COURSE THE ADMISSION WAS INVALID AND EVERYTHING ELSE WAS INVALID."

- The Excel Spreadsheet does not give me access to the data in a format-free way

  "THE ONLY CHALLENGE WE HAVE IS WHEN YOU GO TO THE ERROR REPORTS ON THE REPORTS, YOU KNOW, THEY CREATED A FUN, EASY WAY TO SHOW MAYBE TO A BUSINESS PERSON, BIG GROUPING OF THE CALCULATION BY ME, BY CHAPTER, ETC. BUT WHEN YOU TALKING ABOUT DAY TO DAY PERSON WAS DOING OPERATIONS AND TRYING TO FIGURE THOSE ERRORS, YOU KNOW, YOU GOT TO REMOVE THAT GROUPING THING, REMOVE THE EMPTY COLUMN, CLEAN A BUNCH OF STUFF JUST TO BE ABLE TO WORK ON THE FILE THAT DOESN'T HAVE ANYTHING ELSE BUT THE DATA OF WHAT RECORD HAS BEEN AFFECTED AND WHAT ERROR CODE SO WE WORK ON THE RESOLUTION OF THE ERROR ITSELF. THAT PROCESS ON OUR SIDE IS TAKING EASILY BETWEEN 30-45 MINUTES. JUST TO GET TO WHAT WE NEED TO DO TO START FIXING THE PROBLEMS. AND EVERY TIME IT'S TAKING MORE TIME BECAUSE THERE IS MORE DATA."

- *"WE LOST CONNECTIVITY AGAIN."*
- *"I THINK THAT THE INITIAL UPLOAD IS THE EASY PART. THE FIXING OF ERRORS IS A CHALLENGING ONE AND TIME CONSUMING."*

9.1.11. Interview with Managing Entity (Central Florida Cares Health System, Inc.)

Central Florida Cares Health System, Inc. is one of the Managing Entities that send data to DCF throughout FASAMS. CFCHS serves Brevard, Orange, Osceola and Seminole counties. CFCHS's experience with FASAMS has not been the ideal and here are some of their concerns:

- The process itself is pretty simple. We download the data from our system and create the files that we need to upload them to the FASAMS system.
- Feedback from the tool is not useful to resolve issues
- No way to know what was processed
- No way to pull back out the records already uploaded
- There is a need to reduce the time errors take to be solved
- Export to PDF is not useful
- Export to Excel works but exported file has limitations
- Issue with the file size during upload. It requires to split the load and increases the time.
- UI needs a better approach
- SFTP and Web Services are misused because once a job is complete the error report is not useful.

  "YOU KNOW, HISTORICALLY AND IN THE PREVIOUS SYSTEM, IF THERE WAS AN ERROR, I COULD SEE THE EXACT RECORDS FROM THE SYSTEM THAT WAS ERRORED. SO, I QUICKLY BE ABLE TO GO EXACTLY TO THAT FIELD THAT HAD AN ERROR. IF THERE WERE MULTIPLE ERRORS, THEY WERE GROUPED TOGETHER, THAT PARTICULAR RECORD AND YOU COULD MAKE THOSE FIXES AND MOVE FORWARD."

- It takes a long time to fix errors

  "SO, THE BIGGEST BARRIER RIGHT NOW, THE LARGEST FACTOR, IS THE AMOUNT OF TIME IT TAKES TO GO THROUGH THOSE ERRORS AND ALSO NOT BEING ABLE TO PULL BACK OUT WHAT'S IN THE SYSTEM. SO, WE'RE KIND OF OPERATING BLIND WHERE WE SUBMIT SOMETHING. WE WOULD RESUBMIT EVERYTHING BECAUSE IT'S EXTREMELY CUMBERSOME TO FIGURE OUT WHICH PARTICULAR RECORD FAILED AND THAT'S KIND OF WHAT I'M GETTING AT. SUBMITTED TEN THOUSAND RECORDS AND THERE'S A THOUSAND ERRORS, WE TRY TO FIX AS MANY ERRORS AS POSSIBLE. JUST RESUBMIT IT AND SEE WHAT WE'RE LEFT WITH AND TRY TO NARROW IT DOWN TO WORK WITH IT."

  "WE UPLOADED DATA LAST WEEK AND WAS MORE FOCUSED ON VALIDATION CHANGES ON OUR END, IN ORDER TO REDUCE THE NUMBER OF REJECTED FILES. I ALSO ENCOUNTERED SIGNIFICATE ISSUES WITH INCORRECT FILE UPLOAD REJECTIONS AND SLOWNESS WITH THE FASAMS SITE THAT RESULTED IN MOST OF THE UPLOADS BEING DONE IN THE EVENING."

- The system doesn't handle large files

  "I KNOW THAT THERE'S, THERE'S DIFFICULTIES GETTING LARGER FILE SIZES INTO THE SYSTEM AND WE'VE, WHAT WE'RE SEEING IS THAT IF IT'S GREATER THAN 25 MEGABYTES, THEN WE HAVE TO SPLIT THEM ON OUR END."
  "RIGHT NOW, WITH THE LIMITATIONS OF THE SYSTEM, WE'RE SPENDING BASICALLY TWO DAYS UPLOADING AND THE REST OF THE MONTH TRYING TO GO THROUGH THE ERRORS THAT'S, RETURNED TRYING TO CLEAR THESE UP AND DRILL DOWN TO WHAT THE REAL ISSUES ARE."

9.1.12. Interview with Managing Entity (South Florida Behavioral Health Network, Inc.)

South Florida Behavioral Health Network, Inc. is one of the Managing Entities that send data to DCF through FASAMS. SFBHN serves Miami-Dade and Monroe counties. SFBHN's experience with FASAMS has not been ideal and here are some of their concerns:

- It takes a long time to process files

  "WE GENERATE ANYWHERE, YOU KNOW, 100,000 RECORDS A MONTH IS WHAT WE PROCESSED HERE, SOMETIMES MORE. SO IT TAKES A LONG TIME TO GENERATE THOSE FILES. FIRST OF ALL, BECAUSE OF THE XML SCHEMA CODING. "

- We did not have enough time to create a solution

  "DCF ONLY GAVE US LITERALLY ABOUT THREE AND A HALF, FOUR MONTHS FROM THE TIME THAT WE HAD THE LAST CHAPTER OF INFORMATION FOR THE FASAMS DATA STRUCTURE IN ORDER TO GENERATE RECORDS." "TO BE ABLE TO PRODUCE THE FASAMS RECORDS, WE HAD TO TAKE A LOT OF SHORTCUTS. AND AS A CONSEQUENCE, OF THOSE SHORTCUTS OUR SYSTEM IS FAR FROM BEING EFFICIENT IN HOW IT PROCESSES THE DATA AND HOW IT GENERATES THE DATA."

  "AND SO THERE WERE VALIDATION CODES THAT WE HAD TO WAIT TO IMPLEMENT BECAUSE WE HAD NO IDEA HOW CHAPTER ONE WAS GOING TO INTERRELATE WITH CHAPTER FIVE OR CHAPTER TWO WITH CHAPTER FOUR, YOU KNOW, CHAPTER SIX, YOU KNOW, IF WE JUST, NONE OF THAT INFORMATION WAS AVAILABLE TO US"

- The upload and error correction process take a long time

  "IT'S A VERY LENGTHY PROCESS TO GENERATE THOSE RECORDS AND THEN OF COURSE, TO UPLOAD THEM, ESPECIALLY THE CLOSER WE GET TO THE DEADLINE FOR DATA SUBMISSION EVERY MONTH, IT SOMETIMES COULD TAKE UP TO A DAY FOR US TO HAVE ALL OF OUR RECORDS PROCESSED. SO, FOR THAT 24-HOUR PERIOD, WE DON'T KNOW REALLY WHETHER OR NOT THERE'S ANY ISSUES WITH THE DATA OR WHETHER OR NOT THERE'S GOING TO BE ANY REJECTIONS."

  "WE ENCOUNTER UNEXPECTED ERRORS FREQUENTLY. AND WITH THOSE UNEXPECTED ERRORS, WE HAVE NO WAY OF KNOWING WHAT DATA GOT IN BECAUSE WE DON'T HAVE ANY ACCESS TO SEE THE DATA THAT ACTUALLY MAKES IT INTO FASAMS".

  "I WOULD LOVE TO SEE IT BEING ABLE TO PROCESS RECORDS A LOT FASTER THAN WHAT IT CURRENTLY DOES. UH, HAVING TO WAIT 24 HOURS, YOU KNOW, TO SEE WHETHER OR NOT THAT LAST RECORD THAT WE TRANSMITTED MADE IT, UM, YOU KNOW, SOMETIMES EXTENDS PAST OR BEYOND THE DUE DATE THAT, THAT THE DATA IS, YOU KNOW, BEYOND THE 18TH OF THE MONTH."

  "THERE'S A LOT OF GUESSWORK THAT'S GOING INTO THIS WHOLE PROCESS RIGHT NOW."

  "A LOT OF TIMES WE ENCOUNTER MISMATCHES WITH SOURCE RECORD IDENTIFIERS. SO, IT'S A MATTER OF GOING BACK, REVERIFYING THOSE SOURCE RECORD IDENTIFIER IDENTIFIERS, MAKING SURE THAT THEY'RE LINKING, THE APPROPRIATE WAY. AND THEN THE WHOLE PROCESS THEN OF REGENERATING THE RECORDS, RETRANSMITTING THE RECORDS AND DEPENDING ON THE NUMBER OF RECORDS THERE IS, THERE COULD BE A WAIT IN ORDER FOR EVERYTHING TO PROCESS."

  "TRANSMISSION PROCESS, IT'S JUST VERY CLUNKY. IT'S VERY CUMBERSOME AND RIGHT NOW WE JUST HAVE NO WAY OF, OF PERFORMING ANY LEVEL OF QUALITY ASSURANCE ON WHAT WE SUBMIT. AND SO WHENEVER WE HAVE TO GO BACK TO FIX ANYTHING, IT USUALLY RESULTS IN US HAVING TO RETRANSMIT ALL OF THE DATA THAT WE HAVE PREVIOUSLY TRANSMITTED."

  "WE HAVE TO REPEAT THAT WHOLE PROCESS SOMETIMES TWO, THREE, FOUR TIMES WITHIN A GIVEN MONTH."

  "DEPENDING ON THE RECORD TYPE WE'RE DOING, WE'RE WORKING WITH SOME, THREE, FOUR, FIVE, SIX DIFFERENT LEVELS OF SOURCE RECORDS IDENTIFIER, WHICH MAKES THE PROCESS VERY COMPLICATED."

- FASAMS has increased our workload

  "I HAVE THREE PEOPLE, SO AGAIN, A SMALL STAFF AND OUR WORKLOAD HAS LITERALLY DOUBLED UNDER FASAMS THAN IT DID HISTORICALLY UNDER, UNDER SAMH."

"THE SUBMISSION PROCESS, OUR PROCESS, STARTS ON THE FOURTH OR FIFTH OF THE MONTH. OKAY? AND IT TAKES US BETWEEN THE FOURTH THROUGH THE 18TH, IN ORDER TO GET EVERYTHING SORTED, VALIDATED PROCESSED AND THEN UPLOADED INTO FASAMS. AND THEN ONCE THAT IS DONE, THE REST OF THE MONTH IS BASICALLY JUST TROUBLESHOOTING, ERROR CORRECTING AND REPROCESSING DATA. SO, IT'S A CYCLE THAT DOES NOT END. IT'S LITERALLY THE ENTIRE MONTH, ALL MONTH, 40 HOURS A WEEK, NONSTOP."

"WHEN WE IMPORT DATA INTO OUR SYSTEM, WE TRIED TO MIRROR AS BEST AS POSSIBLE THE VALIDATION, RULES THAT FASAMS HAS LAID OUT."

- We don't know what records were successfully imported

"WE HAVE NO VISIBILITY. SO, THERE'S NO WAY OF US BEING ABLE TO QUERY OR RUN A DOWNLOAD OF WHAT'S IN FASAMS IN ORDER TO COMPARE TO WHAT WE THINK WE SUBMITTED, UM, AND HAVE IN OUR SYSTEM. SO, IT MAKES, IT MAKES THE WHOLE PROCESS OF QUALITY ASSURANCE AND TROUBLESHOOTING TWICE AS CUMBERSOME BECAUSE WE WERE LITERALLY SHOOTING IN THE DARK AND HOPING THAT, UH, WE'RE ABLE TO ISOLATE THE ISSUES AND CORRECT THEM"

"IF YOU TRANSMIT THE RECORDS THROUGH SFTP, BUT THEN YOU HAVE TO LOG INTO THE PORTAL IN ORDER TO GET YOUR RESULTS, IN ORDER TO SEE WHAT GOT ACCEPTED AND IF YOU GOT ANY ERRORS. SO, WITHOUT GETTING THAT IMMEDIATE FEEDBACK, WE JUST PREFER RIGHT NOW THE PORTAL BECAUSE AS YOU PUSHED THE RECORDS IN, YOU SEE THE RESULTS, YOU KNOW, AS SOON AS THE SYSTEM PROCESSES THAT UPLOAD."

"THE SYSTEM IS CAPABLE OF SENDING NOTIFICATIONS. I TURNED IT OFF BECAUSE YOU GET AN EMAIL FOR EVERY SINGLE ERROR AND SO WHAT WAS GOING ON WAS MY INBOX WOULD EXPLODE WITH HUNDREDS OF EMAILS."

"BEING ABLE TO VIEW WHAT HAS, WHAT IS ACTUALLY THERE. BEING ABLE TO SOMEHOW DOWNLOAD IT OR DOWNLOAD A REPRESENTATION OF IT THAT WE WOULD BE ABLE TO THEN, UH, WRITE SCRIPTS THAT WOULD COMPARE THAT AGAINST WHAT WE HAVE SO THAT WE CAN BETTER ISOLATE, UM, THE RECORDS THAT DO HAVE ISSUES OR ERRORS SO THAT WE CAN JUST FOCUS ON THOSE AND ONLY TRANSMIT THOSE ONES, THOSE ARE FIXED."

9.2.Submission Module

Managing entities use the submission module to upload files for import into FASAMS. We heard numerous complaints about the way that the submission process works and were able to verify that these issues have not been resolved. The most critical issues are:

- Process takes a long time. MEs submit files and may have to wait a day before the process finishes.
  - Recommendation: The system should be updated so that files are processed quickly, and submitters get feedback the same day.
- MEs don't know which records made it into the system.
  - Recommendation: Provide a reconciliation report for each file that is submitted for import that clearly states which records were imported successfully and which ones had errors, and what exactly the errors are. This report needs to be exportable so that the MEs can compare the data against their system. Create new reports that the MEs can run on demand and see what records are in the system and which ones were rejected due to errors.
- MEs spend a lot of time fixing errors. There is no easy way to see what the errors are so there is a lot of guesswork to resolving the errors.
  - Recommendation: Creating the reconciliation process should help with this issue.
- MEs can't delete files submitted by mistake.
  - Recommendation: Create a function that allows submissions to be cancelled.

9.2.1.Submission Module Functional Overview

Managing Entities use the submission module to upload files for import into FASAMS. We heard numerous complaints about the process used to submit data to FASAMS, the error resolution and reconciliation process. MEs can submit files using the user interface, placing files on an SFTP site or using web services. The MEs use the user interface because it is the least problematic. Even so, there are still many issues with the submission module user interface. A few of the items are listed in the "Challenges" section below.

The process flow is diagrammed below:

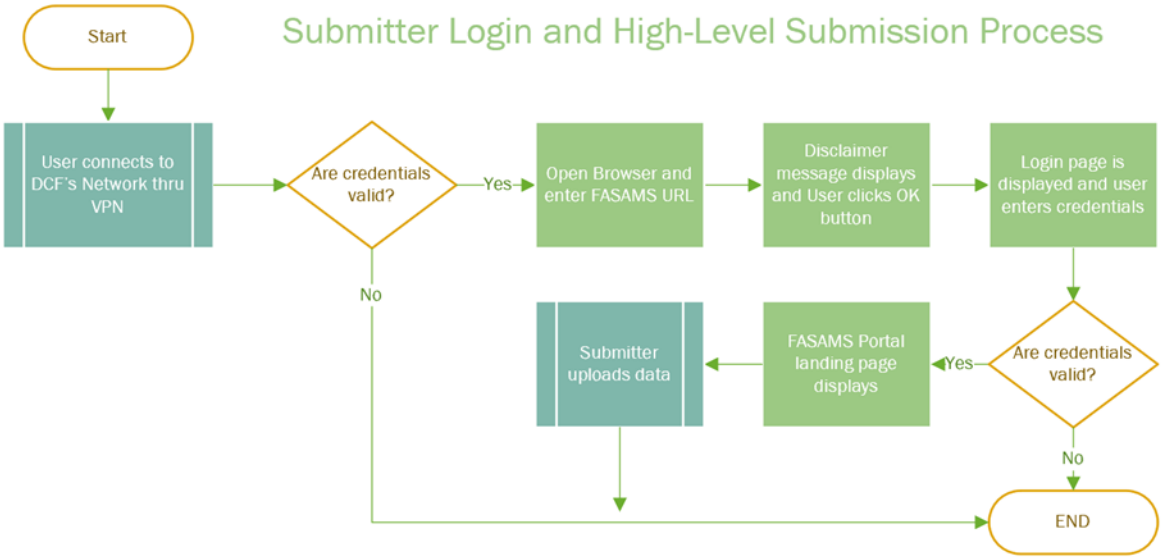


FIGURE 1 - SUBMITTER LOGIN AND HIGH-LEVEL SUBMISSION PROCESS

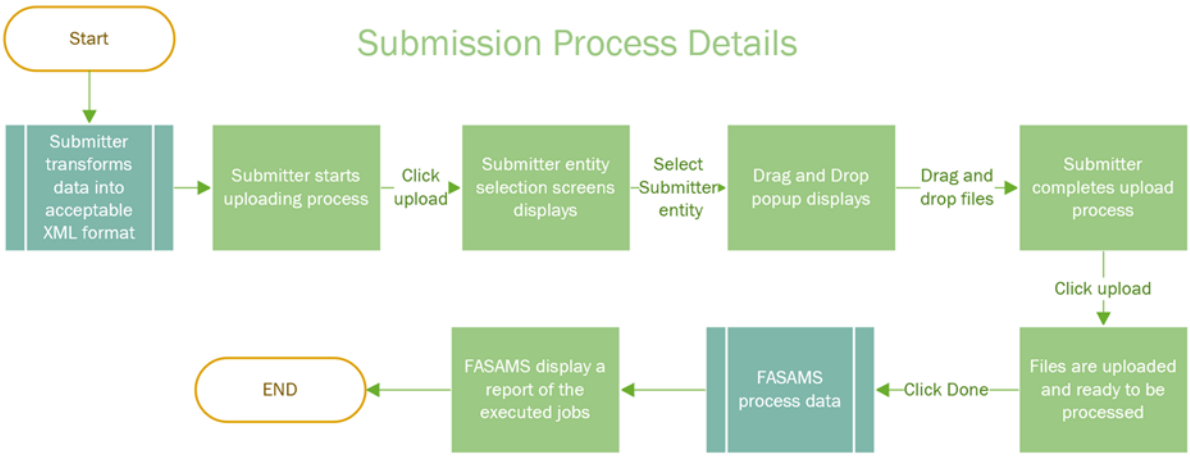


FIGURE 2 - SUBMISSION PROCESS DETAILS

9.2.2. Challenges with the 'Submission' module are as follows:

1. **MEs spend a lot of time fixing errors.** We heard that it takes a long time to identify errors in files that are submitted. One ME stated, "In the previous system, if there was an error, I could see the exact records from the system that was errored. So, I was quickly be able to go exactly to that field that had an error. If there were multiple errors, they were grouped together in that particular record and you could make those fixes and move forward. So, the biggest barrier right now, the largest factor, the amount of time it takes to go through those errors". Another said "the submission process, our process, starts on the fourth or fifth of the month. Okay? And it takes us between the fourth through the 18th, in order to get everything sorted, validated processed and then uploaded into FASAMS. And then once that is done, the rest of the month is basically just troubleshooting, error correcting and reprocessing data. So, it's a cycle that does not end. It's literally the entire month, all month, 40 hours a week, nonstop."

- Recommendation: Provide an easy way for the user to identity the record and field that caused an error.
- Recommendation: Group multiple errors on a record together to make it easier to fix them all at once.

[Of note: The Oz team was told some MEs submit their files to the UAT site repeatedly until they get them to go through with no errors. What this means is that the UAT environment contains actual PHI and not masked or fake data. This may be a HIPAA violation.]

2. **Process takes a long time.** We heard that MEs submit files and then must wait until the next day for the process to finish.
   - Recommendation: The system should be updated so that files are processed quickly, and submitters get feedback the same day.
3. **Large files take a long time.** We heard that large files take a long time to submit – if the file is larger than 25 megabytes, then the MEs must split the file into multiple smaller files and submit them. According to Pamphlet 155-2 Chapter 1 Introduction, FASAMS will accept any file that is smaller than 50MB.
   - Recommendation: The system should be re-architected so that importing a 50MB file takes no longer than one hour.
4. **Users want to see the files that they submitted.** We were told and we saw that the user has no way to view the original file that was submitted.
   - Recommendation: Add the original file that was uploaded with the submission results so that the original file can be viewed.
5. **It's easy to submit files in the wrong order by mistake.** We heard and saw that users may accidentally upload the data files in the wrong order which will cause errors in the import process. For example, if the provider file is submitted after the treatment episodes for that provider, the records in the treatment episode file cannot be imported.
   - Recommendation: Create a way for the user to identify the dataset type for a file when it is uploaded for import. The system should then be smart enough to process the files in the correct order.
6. **Users want a reconciliation process.** We heard that there is not a reconciliation process so that the user can compare the submission outcome against the file that was submitted. An ME stated, "I do not know what records have been accepted by FASAMS as successful." This was reinforced by other MEs when we spoke with them.
   - Recommendation: Provide a report for each file submitted that contains the following (this can be imported into MEs system, if needed for reconciliation):
     - Total number of records in the file
     - Total number of records successfully imported
     - Total number of records that were rejected
     - Total number of records that contained warnings
     - Detail of records successfully imported
     - Detail of records that were rejected
     - Detail of records that contained warnings
   - Recommendation: When viewing a job in submissions the system should display the following:
     - Total number of records in the file (new column). This is not clickable.
     - Total number of records successfully imported (existing column – make it clickable). When the user clicks on the number the system will display the records that were successfully imported. This needs to be downloadable in Excel.
     - Total number of records that were rejected (existing column – make it clickable). When the user clicks on the number the system will display the rejected records. This needs to be downloadable in Excel.
     - Total number of records that contained warnings (existing column – make it clickable). When the user clicks on the number the system will display the records that contain warnings. This needs to be downloadable to Excel.

7. **Users want to be able to remove files submitted by mistake or with errors**. We heard that "not being able to pull back out what's in the system" is an issue for MEs. We also were able to verify that the system does not have this function.
    - Recommendation: Allow users to back out a file that was imported into the system – exploring the impact and the specific business rules would need to be defined to specify what files are eligible to be removed and which ones are not.
8. **Users would like web services to be easier to use.** We heard that Web Services are not being used by the MEs submitting their data, because in order to use the web service to submit files, the submitter must be connected to the VPN. Web services would need to be called by a server and the MEs don't know how to connect the server to the VPN using a backend connection.
    - Recommendation: Add the appropriate security to the web service so that the server does not need to connect to the VPN.
9. **Easy to upload to upload to the wrong customer.** We heard that "If you select the wrong ME, you can upload to the wrong customer".
    - Recommendation: Provide a confirmation screen with the summary of what the user are about to upload and for which ME.
    - Recommendation: Allow users to back out a file that was submitted by mistake.
10. **It's hard to tell what is going on**. When the OZ team attempted to upload files to be processed, the status for each file was set to "initialized" and 24 hours later the status was still set to initialized. The system did not provide feedback on the processing status (if the import started, how many records processed already, etc.). When we asked why the status had not changed, we were told that when files are submitted, they go through the following statuses:
    - Initialized
    - Processing
    - Complete

    And that they would investigate and let us know what is going on. We were told that the file was stuck due to a defect in the UAT site and that it was fixed.
    - Recommendation: Provide runtime feedback when a file is submitted so that the user can tell if the file is being imported, exactly where it is in the process, or if there is an error. In addition, provide a way for the user to cancel a job.
11. **System does not give immediate feedback when there is an error**. When we tried to upload a file that contained an invalid character, the status was set to "initialized" and the system did not provide immediate feedback of the error. When we asked why the file was not processed, we were told that the file contained an invalid character and that the system will prevent the file with the error and any subsequent files from being processed. We were told that the way to cancel the submission is to call DCF and wait for them to investigate. The turnaround time for this process may be more than a day.
    - Recommendation: Provide immediate feedback when a file contains an error.
    - Recommendation: Provide a way for the user to cancel a job.

9.2.3. General User Experience Recommendations for the Submission Module

1. **Observation**: On the Submission page, Jobs are listed in the order that they were submitted. There is no way to filter or group jobs. (see figure-3)
    - Recommendation: Consider adding a filtering functionality that will provide the ability for users to view by various Datasets, as needed as well as a function to group that data.

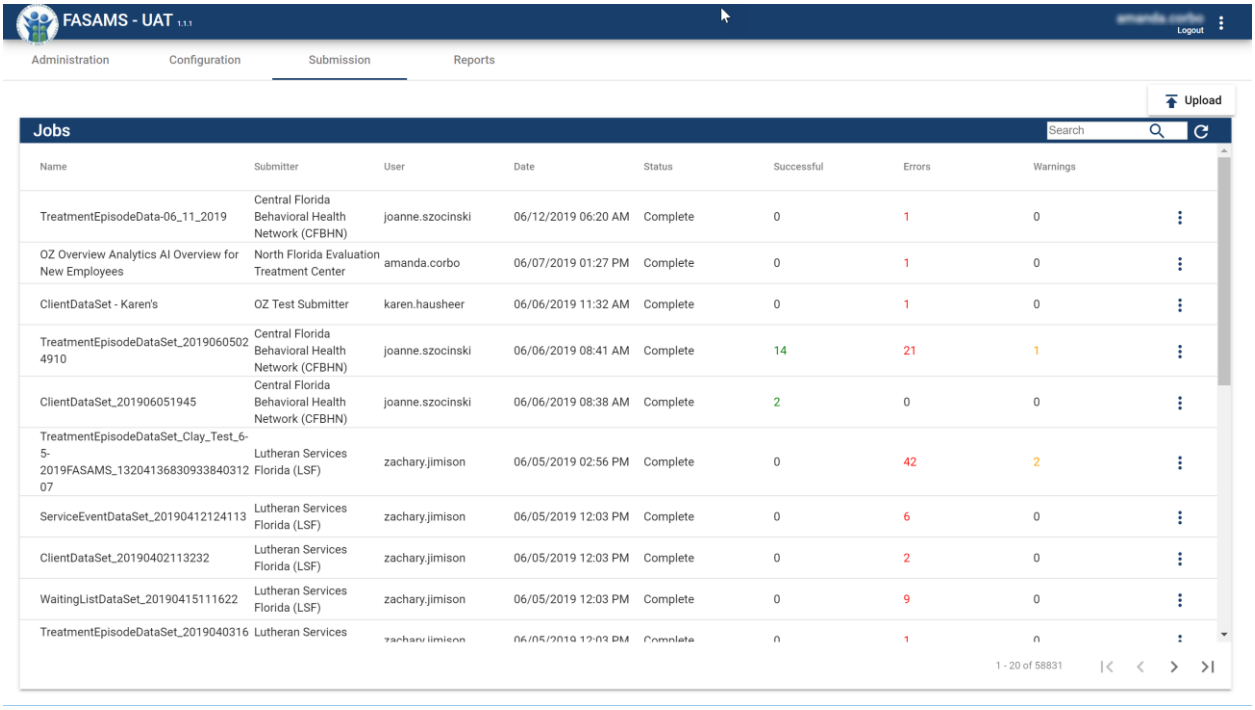


FIGURE 3 - SUBMISSION PAGE JOBS

2. **Observation**: From the submission page, once the user clicks the 'Upload' button, a box appears asking the user to select a submitting entity, once this window appears, there is no way to cancel, close or back out of this action. The user must select a submitter in order to get to the next screen.
    - Recommendation: Light dismiss (clicking on another area on the screen) should be used on this screen to allow the user to switch to a different task without friction OR add an 'X' icon or other close function on this window.

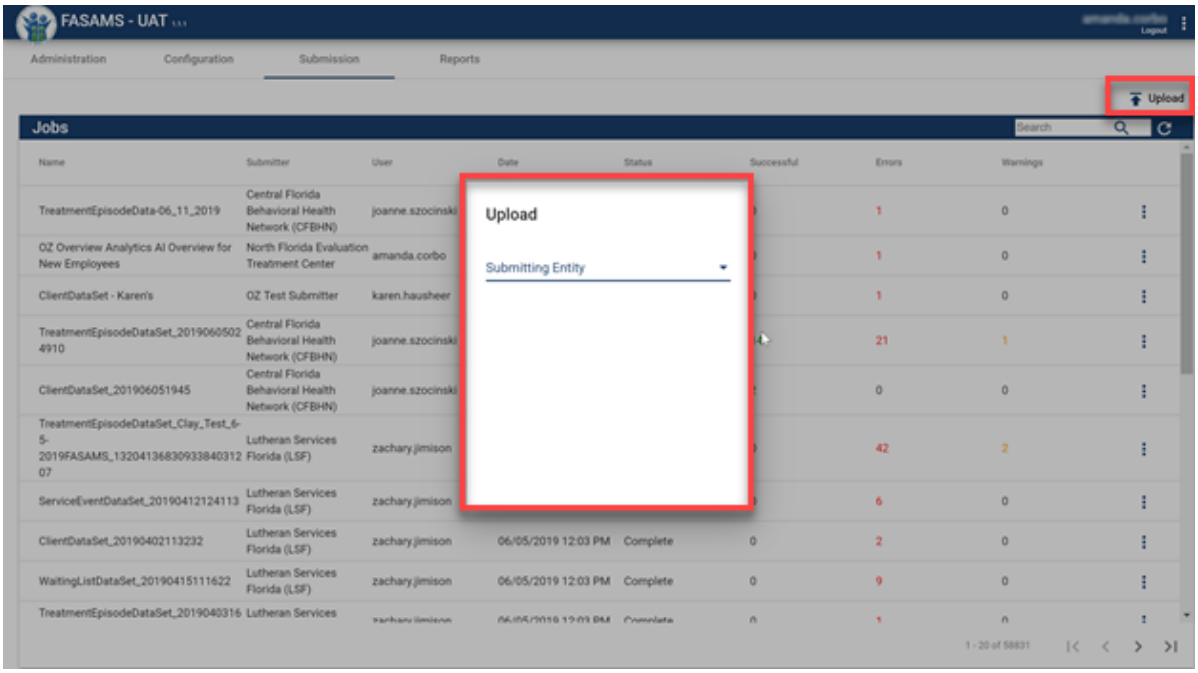


FIGURE 4 - SUBMISSION FILE UPLOAD

Observation: Once the user clicks on the Upload button, there is no way to cancel out of the upload process. It is also not possible to remove files once they are selected.

- Recommendation: Add the ability to cancel the process and remove files from the list.

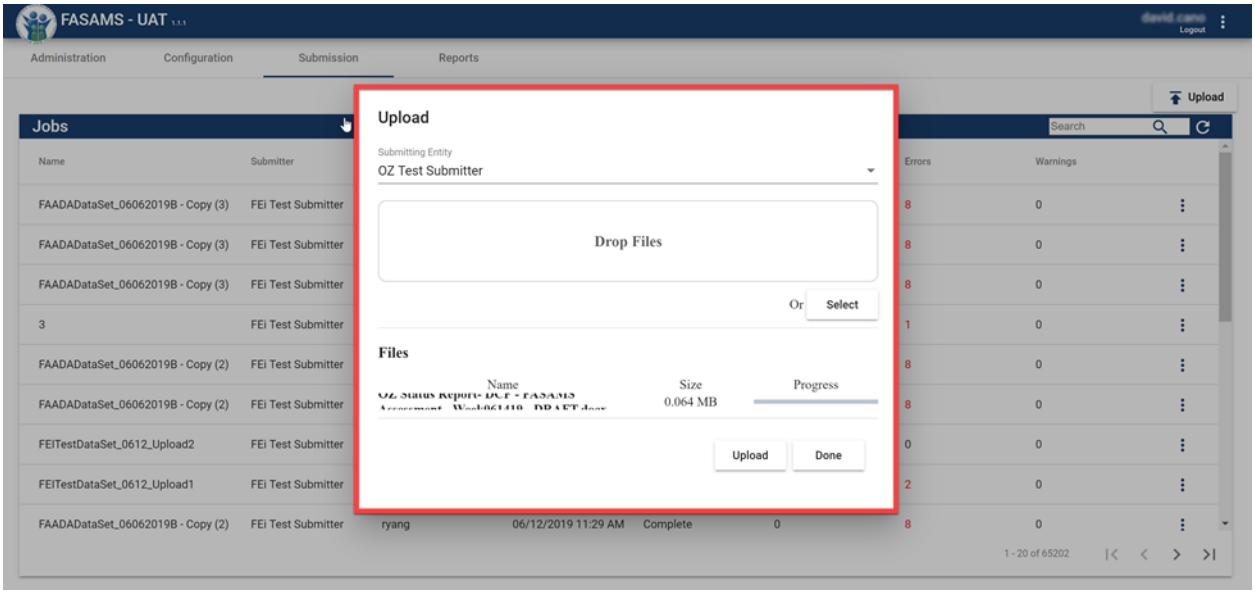


FIGURE 5 - SUBMISSION UPLOAD

9.3.Report Module

9.3.1.Report Module Summary

Twenty-nine standard reports are available in FASAMS. During our interviews we heard that there are issues with the Reports module and we later confirmed that these issues have not been resolved. The most critical issues are:

3. Reports are incorrect. We heard and saw that some reports contain data that appears to be incorrect.
    - Recommendation: Perform a full Quality Assurance Review for each report and correct the identified defects.
4. Report export files are difficult to use. Export does not create files that are usable for analysis or import.
    - Recommendation: Add the ability to save the data and the column headings only.
5. Custom reports are not available. Users can only run standard system reports - custom/ad hoc reporting is not available.
    - Recommendation: Provide the ability to create custom reports.
6. Additional prompts are needed. We heard that additional prompts are needed for the reports. For example, a prompt to filter by providers is not available. Because of this, users need to run reports multiple times in order to get the data that they need
    - Recommendation: Evaluate what additional prompts are needed on reports and add them.

9.3.2.Functional Overview

The system contains a set of standard reports. When users log in, they have an option to run the reports that are available to them based on their security role. Users have reported various issues with the reporting module, including the performance and data accuracy. While the performance issues seem to have improved, the data issues remain.

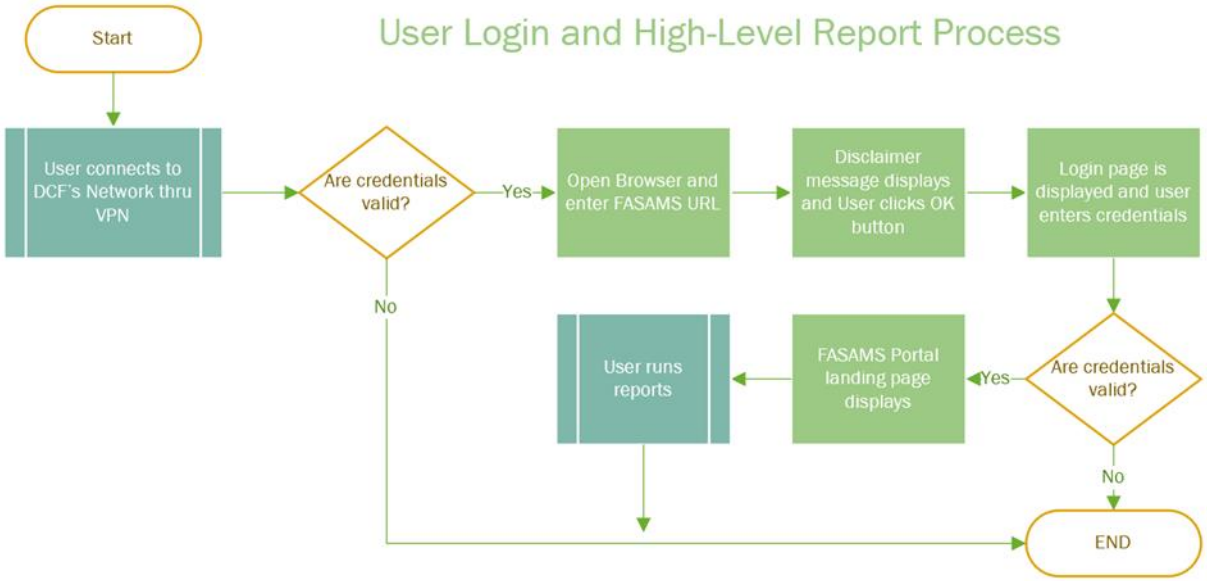


FIGURE 6 - USER LOGIN AND HIGH-LEVEL REPORT PROCESS

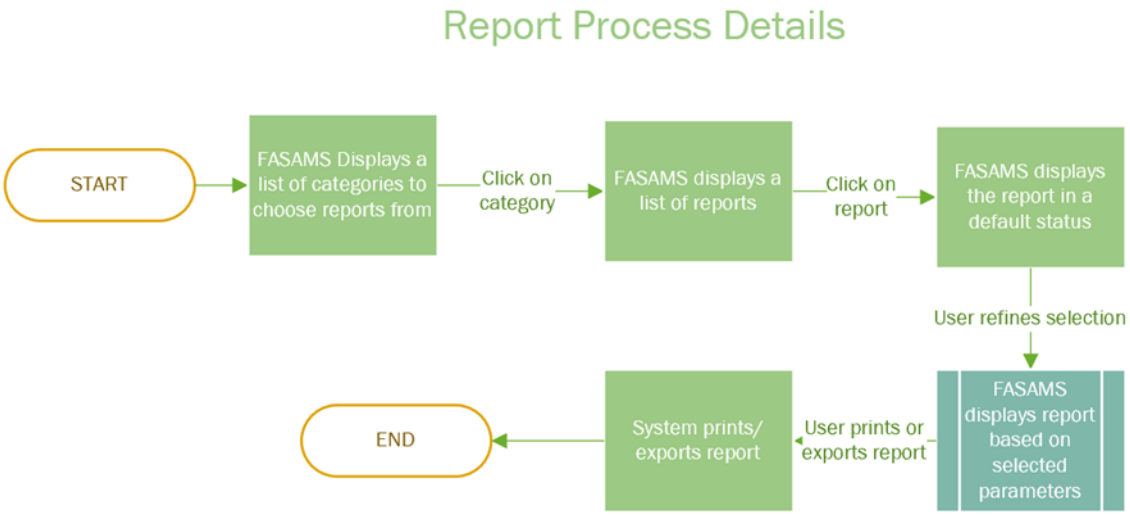


FIGURE 7 - REPORT PROCESS DETAILS

The reports module contains the following reports:

- Acute Care Reports:
  - Indigent Clients Served
  - Operational Beds, Purchased Beds, and Occupied Beds
  - Operational Beds Occupancy Rates
  - DCF Purchased Bed Occupancy Rates
  - Total Licensed Beds
  - SMHTF Operational Bed Counts
- Client
  - Client Search
- Contract Compliance
  - Output Measures – Cumulative
  - Output Measures – Monthly
  - Output Measures – Cumulative Performance
  - Output Measures – Cumulative Data Set Size
  - Output Measures – Monthly Performance
  - Output Measures – Monthly Data Set Size
- Exception
  - 90 Day Performance Outcome Measure Exception
  - CSU Bed Utilization Exception
- Federal
  - SABG 7 – SABG Statewide Entity Inventory
  - URS 5A (MHBG10A) – Profile of Clients by Type of Funding Support
  - URS 5B (MHBG10B) – Profile of Clients by Type of Funding Support
  - URS 6 (MHBG 11) - Profile of Client Turnover
  - SABG Table 10 - Treatment Utilization Matrix
  - SABG 11 - Unduplicated Count of Persons Served for Alcohol and Other Drug Use
  - URS 12 (MHBG 12) - State Mental Health Agency Profile
  - URS 16 and 17 (MHBG 19 and 20) - Profile of SMI Receiving Specific Services

- Financial
  - Provider Expenditure Validation (For DCF)
  - Provider Expenditure Validation (For Quality Assurance Purposes)
- Job Submission Performance
  - Failed Records
  - Overall Job Performance
  - Submission Summary
  - Submission Rollup

9.3.3. Report Module Challenges

During our review, we saw and heard that there are shortcomings with the Reports module. The area of greatest concern is that the data that the reports are displaying is incorrect. Other challenges are: users see an error message when opening a report exported to excel, users would like to be able to create custom/ad-hoc reports, they would like additional standard reports created, have additional prompts, the ability to select multiple values for all parameters and improved export functionality. The usability review uncovered areas for opportunity to improve navigation, apply proper style guides and standards to increase usability and well as readability of the system pages, create more user friendly report names, ensure that when viewing reports within FASAMS; column headings remain visible while scrolling , utilize the entire width of the screen for reports, add the ability to launch a separate window to view a report, and the ability to save custom created reports to your profile to make it easier to re-run custom reports.

The challenges with the reporting module are as follows:

1. **Reports are incorrect.** We heard that some reports contain data that appears to be incorrect. For example, the OZ team was told that the Cumulative Performance Report contains values of 13.87% when the target is 95%.
   - Recommendation: Perform a full Quality Assurance Review for each report to ensure that what is appearing on the report matches the requirements. Requirements may also need to be reviewed to confirm that they make sense. Once the review is complete, identify the cause and make the appropriate corrections.
2. **Reports are incorrect.** We heard that some reports contain data that appears to be incorrect. The cumulative performance report shows a total YTD amount contracted in excess of $5B, when the budget for the department is less than that. So, it does not seem plausible that the report is correct.
   - Recommendation: Same as #1

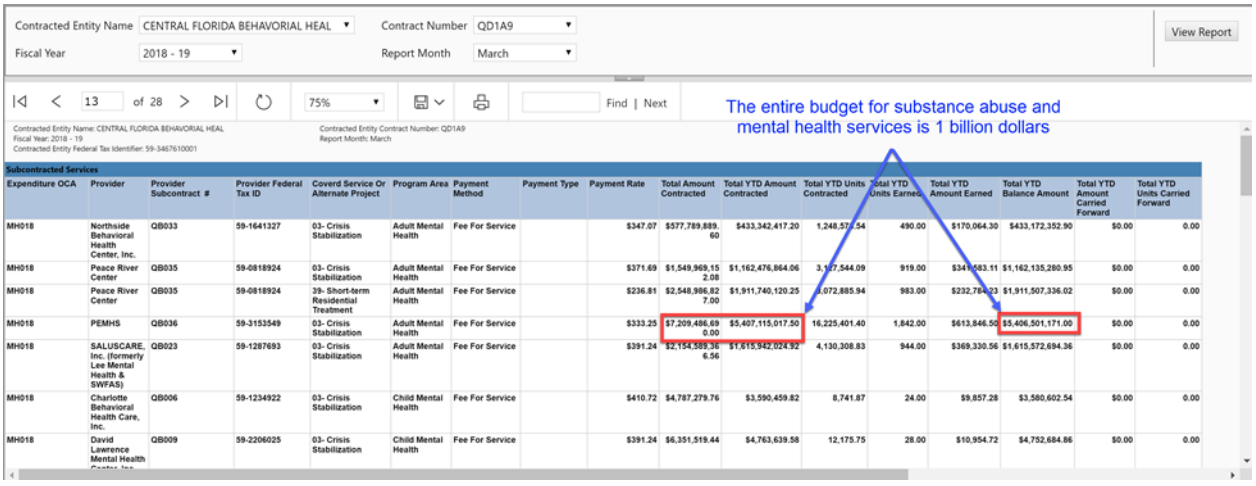
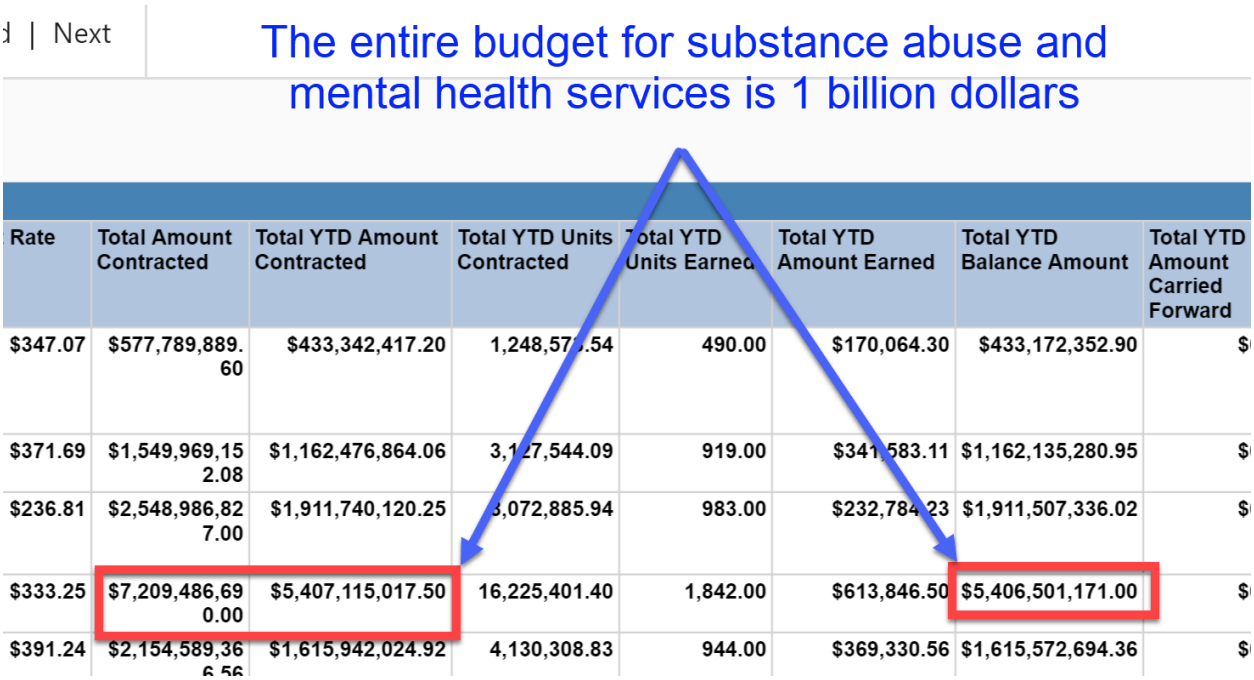


FIGURE 8 - CUMULATIVE PERFORMANCE REPORT (ZOOMED OUT)



Rate	Total Amount Contracted	Total YTD Amount Contracted	Total YTD Units Contracted	Total YTD Units Earned	Total YTD Amount Earned	Total YTD Balance Amount	Total YTD Amount Carried Forward
$347.07	$577,789,889.60	$433,342,417.20	1,248,57[illegible].54	490.00	$170,064.30	$433,172,352.90	$
$371.69	$1,549,969,152.08	$1,162,476,864.06	3,1[illegible]7,544.09	919.00	$341,[illegible]583.11	$1,162,135,280.95	$
$236.81	$2,548,986,827.00	$1,911,740,120.25	[illegible],072,885.94	983.00	$232,784.23	$1,911,507,336.02	$
$333.25	$7,209,486,690.00	$5,407,115,017.50	16,225,401.40	1,842.00	$613,846.50	$5,406,501,171.00	$
$391.24	$2,154,589,366.56	$1,615,942,024.92	4,130,308.83	944.00	$369,330.56	$1,615,572,694.36	$

FIGURE 9 - CUMULATIVE PERFORMANCE REPORT (ZOOMED IN)

3. **Custom reports are not available.** We saw and heard that custom reports are not available. Users can run standard system reports that exist in the system that are listed in the reports description above. Custom/ad hoc reporting is not available in the system.
    - Recommendation: Provide the ability to create a custom report.
        - User should be able to select at a field level
        - User should be able to save the custom report in the end user profile
        - Ability to share custom created reports with other end users (admin level)

4. **Additional prompts are needed.** We heard that additional prompts are needed for the reports. For example, a prompt to filter by providers is not available. Because of this, users need to run reports multiple times in order to get the data that they need. For example, the Indigent Clients Served Report has the following prompts:

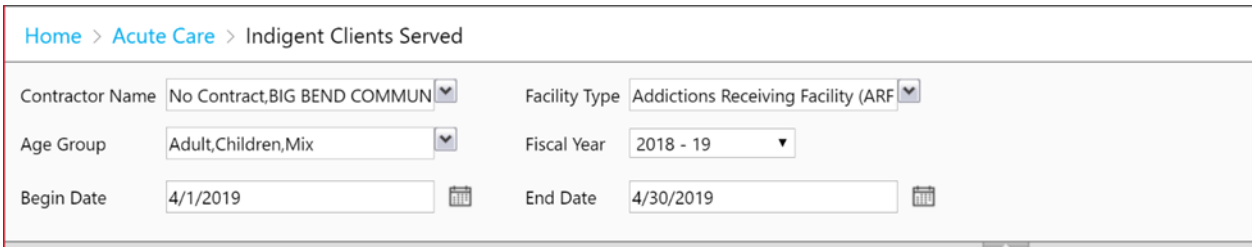


FIGURE 10 - REPORT PROMPTS

    - Recommendation: Evaluate what additional prompts are needed on reports and add them.

5. **Report export files are difficult to use**. We saw that the 'Save' Report Function – The save icon allows the user to export the report into several formats. The issue is that the report that is visible on the screen is exported exactly; including any formatting such as column headings, paging, page numbers, etc. This is great if the user just needs a copy saved but it's not useful if the user needs to manipulate the data. The

MEs end up having to export to excel and remove the formatting manually so that they can use the data for use with their systems or analysis.

- Recommendation: Add the ability to save the data and the column headings only. When the user chooses to export the data, provide a version of the report that contains only the headings and actual data (no page numbers, repeated headings, etc.). Formatting should be stripped (except for dates, phones numbers, etc.).

6. **Need reconciliation report**. We saw and heard that the Failed Records Report and the information provided on the Jobs Screen cannot be used by the MEs for reconciliation because it does not list the items that were imported successfully. In addition, the items that were rejected with errors are not in a format that the MEs can export and import into their system for comparison. The MEs need to see which records were accepted and which records were rejected because of errors in a format that is easy to read and able to be imported into another system.

The Jobs Screen shows the following:

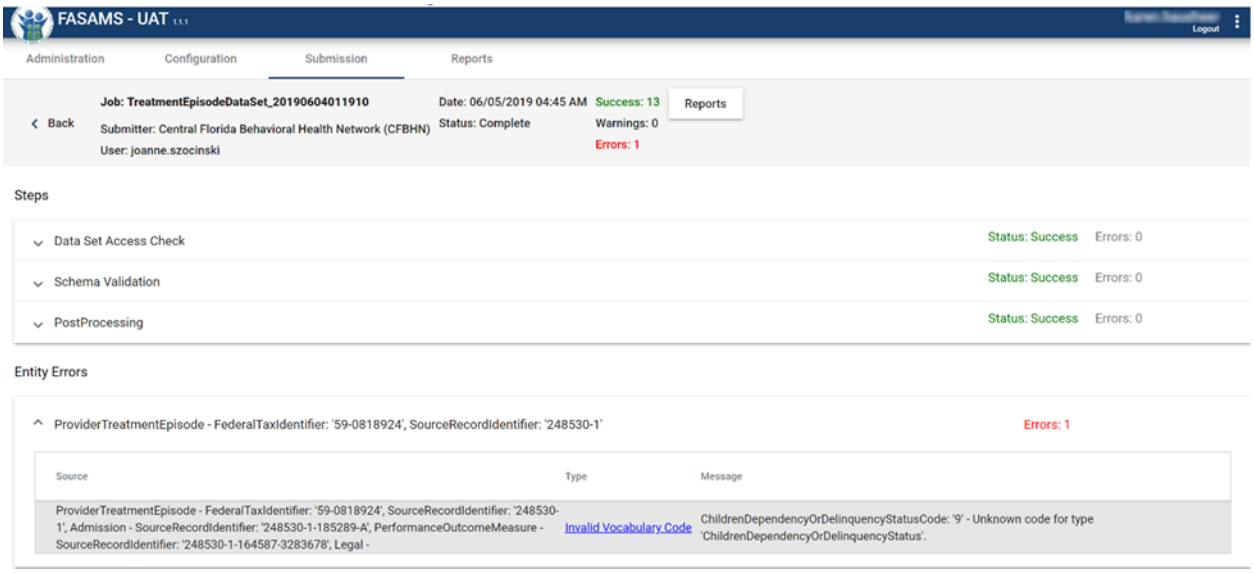


FIGURE 11 - JOB SCREEN

And the PDF and Excel report list the records that contain errors. The 'Failed Records Report' can be run by date range and there is no prompt for the 'Job Name'. The information provided does not contain the records that were imported successfully.

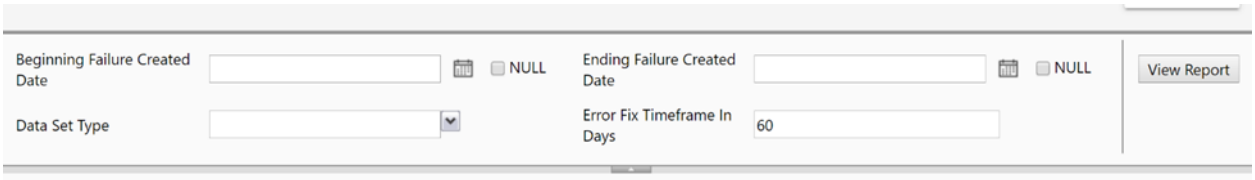


FIGURE 12 - FAILED RECORDS REPORT

- Recommendation: Provide a report for each file submitted that contains the following (this can be imported into MEs system, if needed for reconciliation):
  - Total number of records in the file
  - Total number of records successfully imported

- Total number of records that were rejected
- Total number of records that contained warnings
- Detail of records successfully imported
- Detail of records that were rejected
- Detail of records that contained warnings

7. **Headings need improvement.** We heard that the report headings are not in business language.
    - Recommendation: Update report descriptions, prompts and headings to use correct business language.

9.3.4. Report Module General User Experience Recommendations

The OZ team evaluated the user interface for reports from a user experience perspective and identified the following common areas and opportunities for improvement with the report module.

- **Path:** Reports > Report Viewing Area (Figure 12, see the gray area in screen shot)
- **Observation:** Report Viewing area should be larger to increase the readability of the reports
    - Recommendation: Consider relocating the report 'Categories' selection functionality located in the left-hand pane to another area in order to utilize the full width of the page to view the report after it is run.

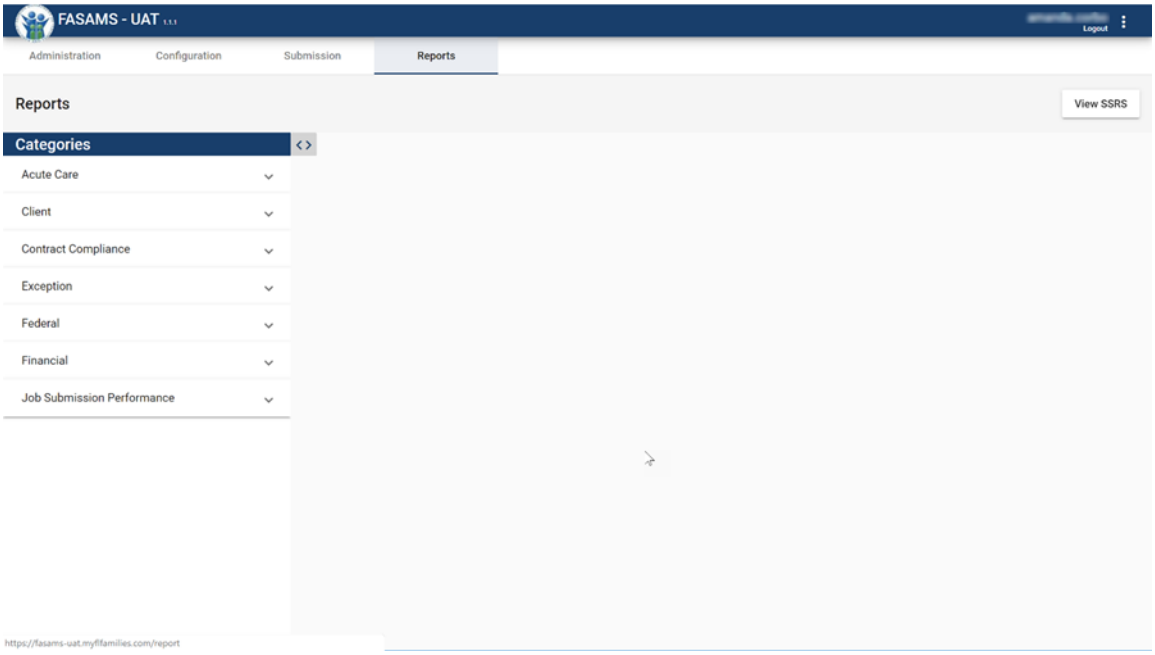


FIGURE 13 - REPORTS LANDING PAGE

- **Path:** Reports > Client Search
- **Observation:** Report names are not consistently shown on the report results view - The report results in the right pane often do not show the name of the report. In the example below, it is difficult to know which report the user has selected.
    - Recommendation: As standard practice the name of the report should be listed at the top of the page so that it is clear what report the user is running. Highlighting the report name on the left navigation menu will help as well.

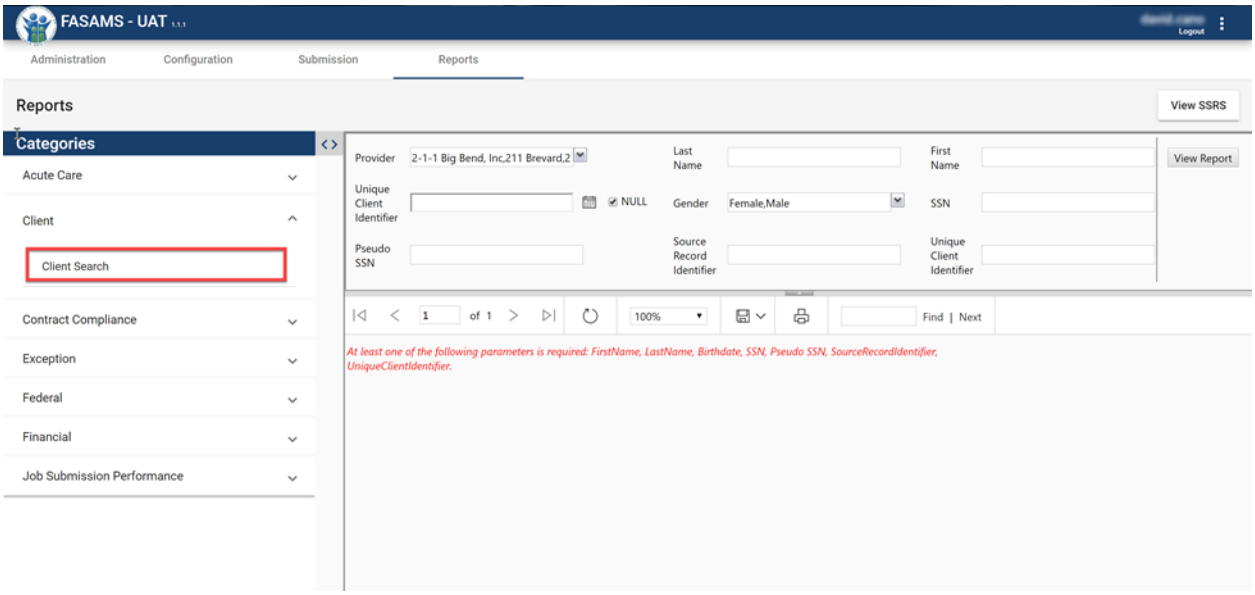


FIGURE 14 - CLIENT SEARCH

- **Path:** Report Acute Care > Operational Beds, Purchased Beds, and Occupied Beds
- **Observation:** Drop down values do not display completely
    - Recommendation: The width of the field drop down window should be widened so that the end user can read the value completely without having to use horizontal scroll bar on the bottom.

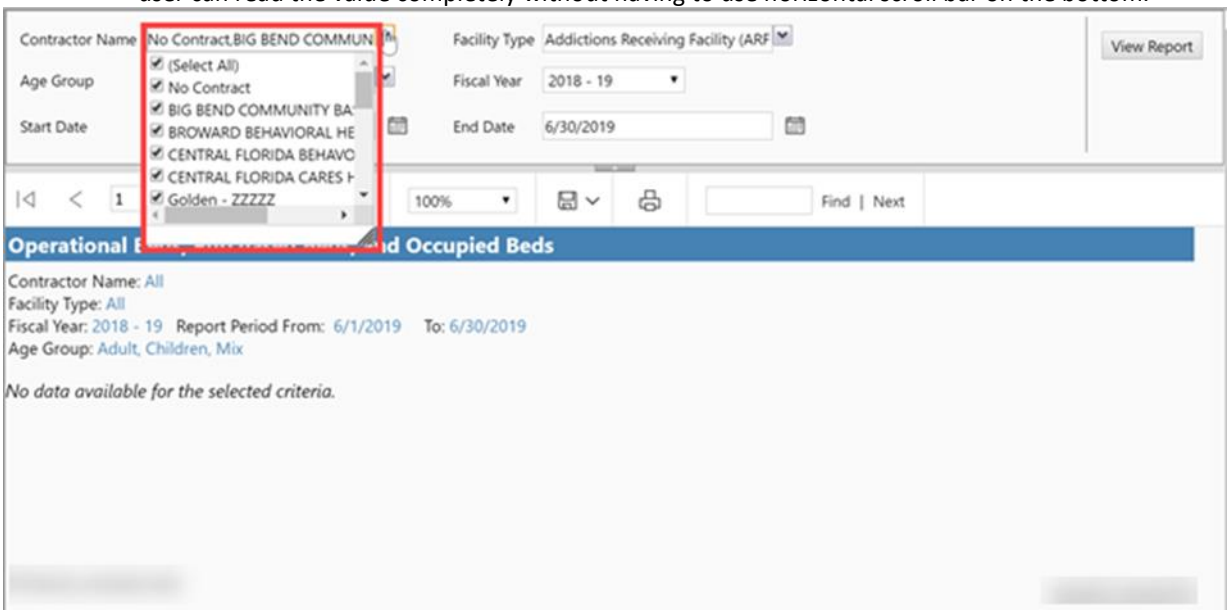


FIGURE 15 - DROP DOWN VALUE NAME CUTOFF

- **Path:** Report > Acute Care > Operational Beds, Purchased Beds, and Occupied Beds
- **Observation:** User has to use the vertical scroll bar to see more than six (6) options.
    - Recommendation: The length of field and drop-down window should be lengthened so that the end user can read the value completely without having to scroll from the top to the bottom.

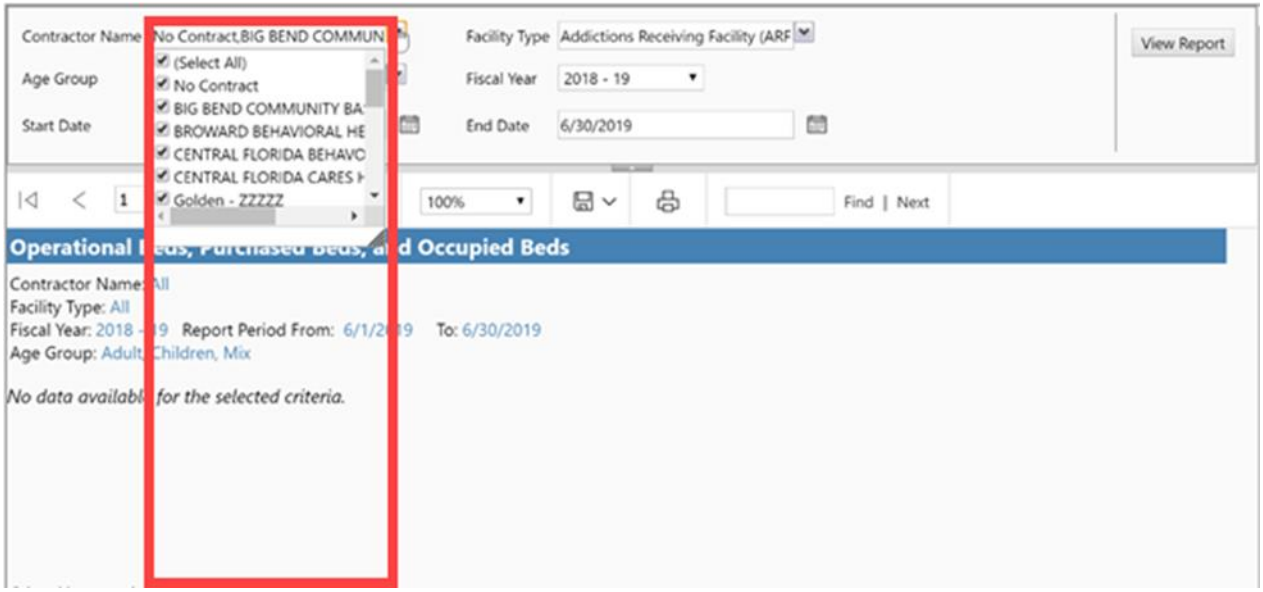
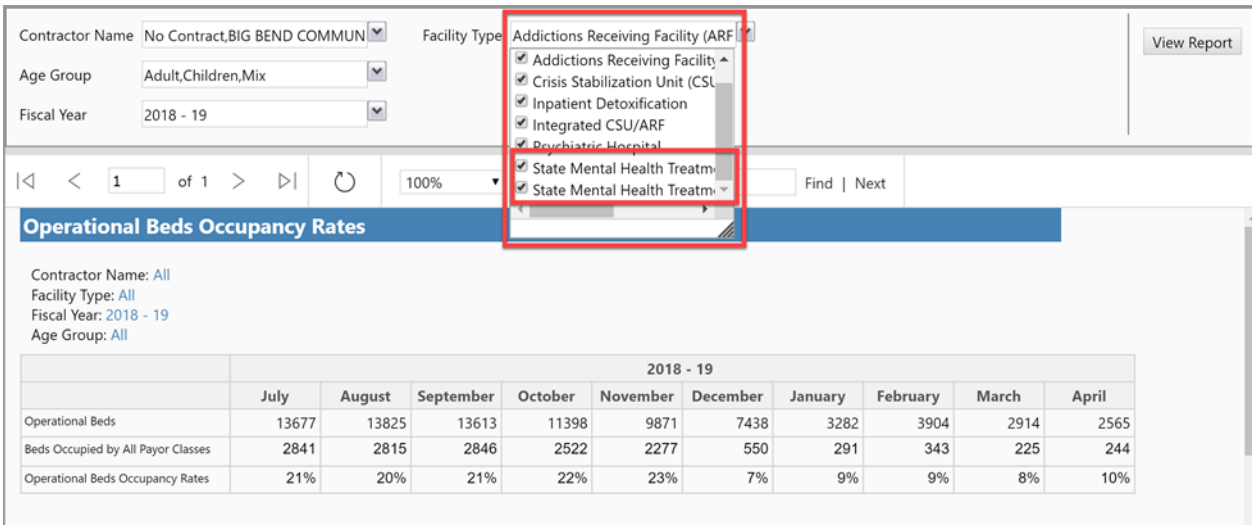


FIGURE 16 - DROP DOWN WINDOW TOO SHORT

- **Path:** Report > Acute Care > Operational Beds, Purchased Beds, and Occupied Beds
- **Observation:** Drop down field value should be easier to find the desired value. Currently, the user must scroll through a very small window in order to locate the value that is desired.
    - Recommendation: Employ a searchable drop-down list in reports.



	2018 - 19									
	July	August	September	October	November	December	January	February	March	April
Operational Beds	13677	13825	13613	11398	9871	7438	3282	3904	2914	2565
Beds Occupied by All Payor Classes	2841	2815	2846	2522	2277	550	291	343	225	244
Operational Beds Occupancy Rates	21%	20%	21%	22%	23%	7%	9%	9%	8%	10%

FIGURE 17 - DROP DOWN VALUE SCROLLING (ZOOMED OUT)

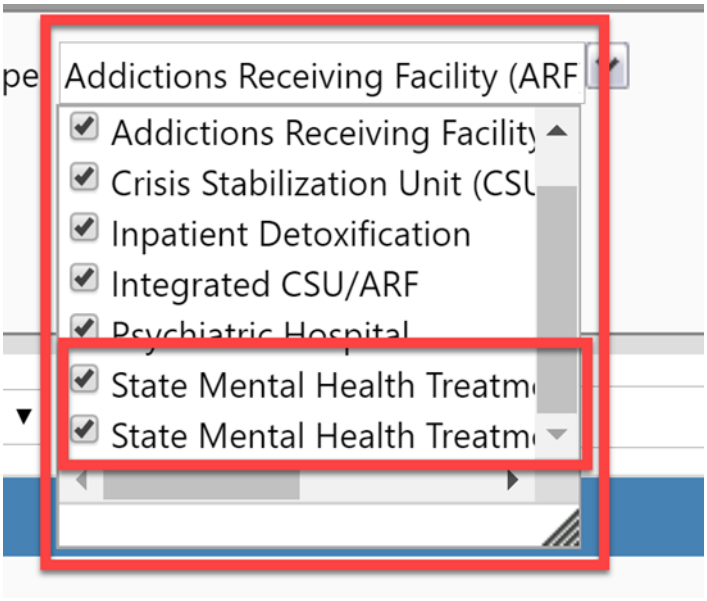
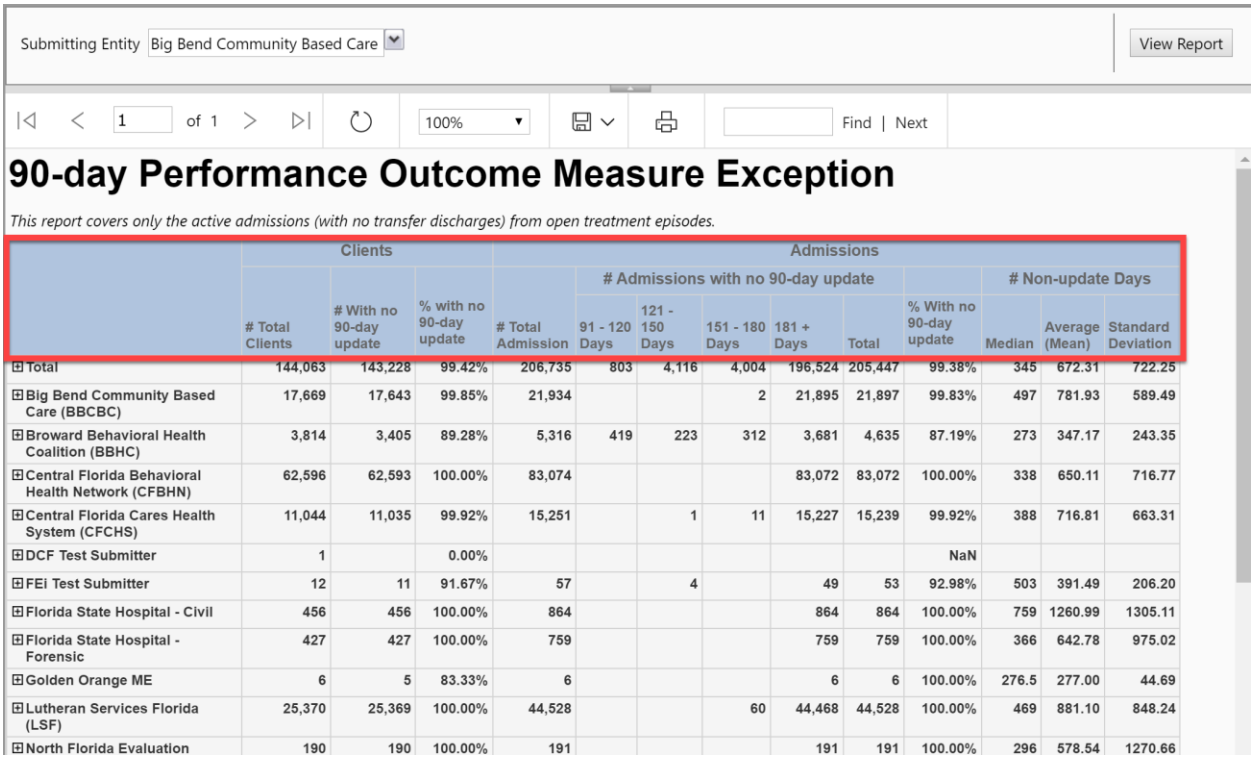


FIGURE 18 - DROP DOWN VALUE SCROLLING (ZOOMED IN)

Path: Reports > Exception > 90-day Performance Outcome Measure Exception

- **Observation:** When scrolling down a report, column headers do not remain in view. Top of page with headings (See Figure 18 & 19).
    - Recommendation: Keep column headers visible when scrolling through data on any given report.

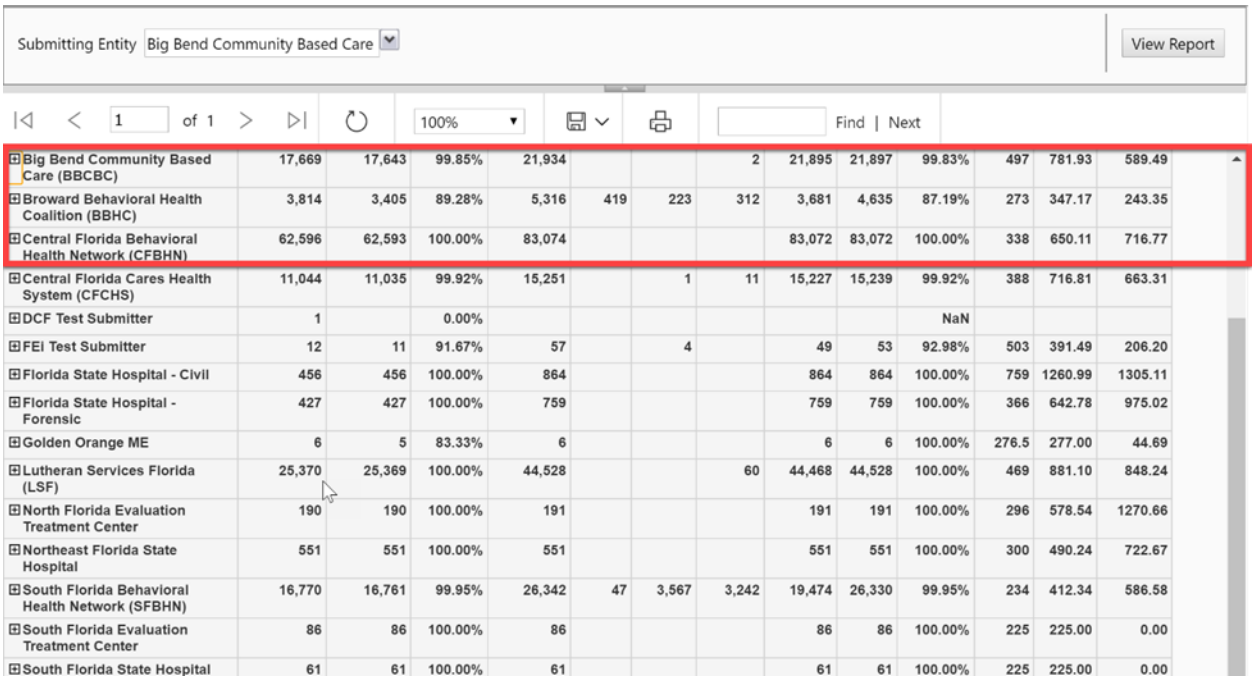


90-day Performance Outcome Measure Exception

This report covers only the active admissions (with no transfer discharges) from open treatment episodes.

	Clients			Admissions								
					# Admissions with no 90-day update						# Non-update Days	
	# Total Clients	# With no 90-day update	% with no 90-day update	# Total Admission	91 - 120 Days	121 - 150 Days	151 - 180 Days	181 + Days	Total	% With no 90-day update	Median	Average (Mean)	Standard Deviation
⊞Total	144,063	143,228	99.42%	206,735	803	4,116	4,004	196,524	205,447	99.38%	345	672.31	722.25
⊞Big Bend Community Based Care (BBCBC)	17,669	17,643	99.85%	21,934			2	21,895	21,897	99.83%	497	781.93	589.49
⊞Broward Behavioral Health Coalition (BBHC)	3,814	3,405	89.28%	5,316	419	223	312	3,681	4,635	87.19%	273	347.17	243.35
⊞Central Florida Behavioral Health Network (CFBHN)	62,596	62,593	100.00%	83,074				83,072	83,072	100.00%	338	650.11	716.77
⊞Central Florida Cares Health System (CFCHS)	11,044	11,035	99.92%	15,251		1	11	15,227	15,239	99.92%	388	716.81	663.31
⊞DCF Test Submitter	1		0.00%							NaN			
⊞FEI Test Submitter	12	11	91.67%	57		4		49	53	92.98%	503	391.49	206.20
⊞Florida State Hospital - Civil	456	456	100.00%	864				864	864	100.00%	759	1260.99	1305.11
⊞Florida State Hospital - Forensic	427	427	100.00%	759				759	759	100.00%	366	642.78	975.02
⊞Golden Orange ME	6	5	83.33%	6				6	6	100.00%	276.5	277.00	44.69
⊞Lutheran Services Florida (LSF)	25,370	25,369	100.00%	44,528			60	44,468	44,528	100.00%	469	881.10	848.24
⊞North Florida Evaluation	190	190	100.00%	191				191	191	100.00%	296	578.54	1270.66

FIGURE 19 - PAGE BEFORE SCROLLING (1 OF 2)



Big Bend Community Based Care (BBCBC)	17,669	17,643	99.85%	21,934			2	21,895	21,897	99.83%	497	781.93	589.49
Broward Behavioral Health Coalition (BBHC)	3,814	3,405	89.28%	5,316	419	223	312	3,681	4,635	87.19%	273	347.17	243.35
Central Florida Behavioral Health Network (CFBHN)	62,596	62,593	100.00%	83,074				83,072	83,072	100.00%	338	650.11	716.77
Central Florida Cares Health System (CFCHS)	11,044	11,035	99.92%	15,251		1	11	15,227	15,239	99.92%	388	716.81	663.31
DCF Test Submitter	1		0.00%							NaN			
FEI Test Submitter	12	11	91.67%	57		4		49	53	92.98%	503	391.49	206.20
Florida State Hospital - Civil	456	456	100.00%	864				864	864	100.00%	759	1260.99	1305.11
Florida State Hospital - Forensic	427	427	100.00%	759				759	759	100.00%	366	642.78	975.02
Golden Orange ME	6	5	83.33%	6				6	6	100.00%	276.5	277.00	44.69
Lutheran Services Florida (LSF)	25,370	25,369	100.00%	44,528			60	44,468	44,528	100.00%	469	881.10	848.24
North Florida Evaluation Treatment Center	190	190	100.00%	191				191	191	100.00%	296	578.54	1270.66
Northeast Florida State Hospital	551	551	100.00%	551				551	551	100.00%	300	490.24	722.67
South Florida Behavioral Health Network (SFBHN)	16,770	16,761	99.95%	26,342	47	3,567	3,242	19,474	26,330	99.95%	234	412.34	586.58
South Florida Evaluation Treatment Center	86	86	100.00%	86				86	86	100.00%	225	225.00	0.00
South Florida State Hospital	61	61	100.00%	61				61	61	100.00%	225	225.00	0.00

FIGURE 20 - PAGE AFTER SCROLLING (2 OF 2)

9.3.5. Report Execution Samples

We heard during our meetings and interviews, that reports took a long time to run. We ran and timed each of the reports three times in the UAT environment in order to assess the performance of the reports. The first run used the default parameters, if present. The second run used an expanded selection criterion so that the report would return more data than the first run. Before we ran the report a third time, we exited the report and then went back into the report so that we could see if there were any issues inherent in leaving that area of the system and returning to the report.

The summary of findings is in the table below. A detailed description of the report, the parameters used for each report, the timing for each run, screen prints of the parameters page, screen prints of a sample of the report, and observations from a user experience perspective are included in the section below.

Report Category	Report Name	1st Run	2nd Run	3rd Run
Acute Care	Indigent Clients Served	3 seconds	3 seconds	3 seconds
Acute Care	Operational Beds, Purchased Beds, and Occupied Beds	4 seconds	4 seconds	4 seconds
Acute Care	Operational Beds Occupancy Rates	3 seconds	4 seconds	4 seconds
Acute Care	DCF Purchased Bed Occupancy Rates	4 seconds	5 seconds	5 seconds
Acute Care	Total Licensed Beds	4 seconds	5 seconds	5 seconds

Report Category	Report Name	1st Run	2nd Run	3rd Run
Acute Care	SMHTF Operational Bed Counts	2 seconds	2 seconds	2 seconds
Client	Client Search	7 seconds	7 seconds	8 seconds
Contract Compliance	Output Measures – Cumulative	7 seconds	7 seconds	7 seconds
Contract Compliance	Output Measures – Monthly	7 seconds	7 seconds	7 seconds
Contract Compliance	Output Measures – Cumulative Performance	7 seconds	7 seconds	7 seconds
Contract Compliance	Output Measures – Cumulative Data Set Size	7 seconds	7 seconds	7 seconds
Contract Compliance	Output Measures - Monthly Performance	7 seconds	7 seconds	7 seconds
Contract Compliance	Output Measures – Monthly Data Set Size	7 seconds	7 seconds	7 seconds
Exception	90 Day Performance Outcome Measure Exception	5 seconds	4 seconds	4 seconds
Exception	CSU Bed Utilization Exception	4 seconds	4 seconds	4 seconds
Federal	SABG 7 – SABG Statewide Entity Inventory	4 seconds	3 seconds	3 seconds
Federal	URS 5A (MHBG10A) – Profile of Clients by Type of Funding Support	4 seconds	3 seconds	3 seconds
Federal	URS 5B (MHBG10B) – Profile of Clients by Type of Funding Support	4 seconds	3 seconds	3 seconds
Federal	URS 6 (MHBG 11) - Profile of Client Turnover	4 seconds	3 seconds	3 seconds
Federal	SABG Table 10 - Treatment Utilization Matrix	3 seconds	3 seconds	3 seconds
Federal	SABG 11 - Unduplicated Count of Persons Served for Alcohol and Other Drug Use	4 seconds	2 seconds	2 seconds
Federal	URS 12 (MHBG 12) - State Mental Health Agency Profile	4 seconds	2 seconds	2 seconds
Federal	URS 16 and 17 (MHBG 19 and 20) - Profile of SMI Receiving Specific Services	4 seconds	2 seconds	2 seconds
Financial	Provider Expenditure Validation (For DCF)	6 seconds	48 seconds	40 seconds
Financial	Provider Expenditure Validation (For Quality Assurance Purposes)	6 seconds	3 seconds	8 seconds
Job Submission Performance	Failed Records	28 seconds	Timed out at 8 mins	28 seconds
Job Submission Performance	Overall Job Performance	1 second	5 seconds	12:33 minutes

Report Category	Report Name	1st Run	2nd Run	3rd Run
Job Submission Performance	Submission Summary	35 seconds	25 seconds	35 seconds
Job Submission Performance	Submission Rollup	8:30 minutes	8:06 minutes	8:30 minutes

Acute Care Reports

The FASAMS system provides five Acute Care reports and allows DCF staff to export them for monthly Acute Care statistics to be published on the DCF website. The five reports are:

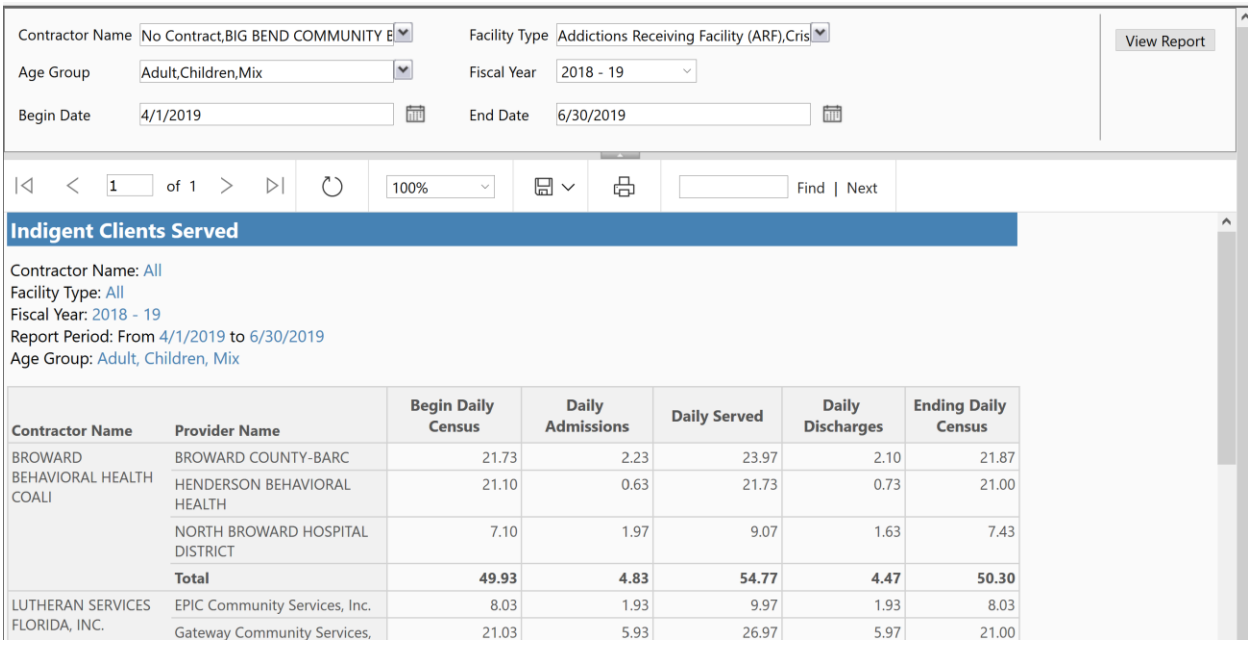
- Indigent Clients Served
- Operational Beds, Purchased Beds, and Occupied Beds
- Operational Bed Occupancy Rates
- DCF Purchased Bed Occupancy Rates
- Total Licensed Beds

Indigent Clients Served Report

The purpose of this report is to provide, in a tabular form, the daily census of indigent clients served during the selected reporting period.

Details of the three runs:

- First run: All options selected for multiselect fields, Response time (3 secs).
- Second run: All options selected for multiselect fields with the largest time frame possible selected, Response time (3 secs).
- Third run after closing the app and coming back: All options selected for multiselect fields with the biggest time frame possible selected, Response time (3 secs).



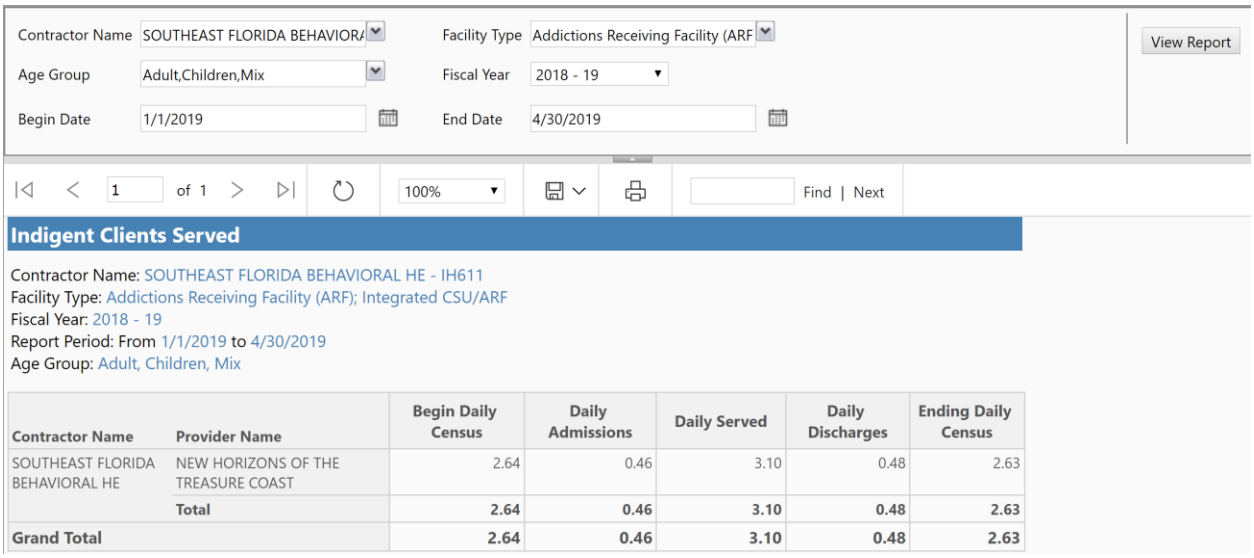
Contractor Name	Provider Name	Begin Daily Census	Daily Admissions	Daily Served	Daily Discharges	Ending Daily Census
BROWARD BEHAVIORAL HEALTH COALI	BROWARD COUNTY-BARC	21.73	2.23	23.97	2.10	21.87
	HENDERSON BEHAVIORAL HEALTH	21.10	0.63	21.73	0.73	21.00
	NORTH BROWARD HOSPITAL DISTRICT	7.10	1.97	9.07	1.63	7.43
	Total	**49.93**	**4.83**	**54.77**	**4.47**	**50.30**
LUTHERAN SERVICES FLORIDA, INC.	EPIC Community Services, Inc.	8.03	1.93	9.97	1.93	8.03
	Gateway Community Services,	21.03	5.93	26.97	5.97	21.00

FIGURE 21 - INDIGENT CLIENTS SERVED REPORT EXECUTION SAMPLE

Observations:

- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.
- There is no option to select multiple fiscal years.
- Choosing the fiscal year does not further filter the 'Begin Date' and 'End Date'. If incorrect dates are entered, the report will display a message telling the user that invalid dates were entered. The system should not allow the user to enter invalid dates in the first place.
- No validation for Start and End date (End date should be greater or equal than start date).
- There is no way to select a provider in the report's parameters.

Sample report:



Contractor Name	Provider Name	Begin Daily Census	Daily Admissions	Daily Served	Daily Discharges	Ending Daily Census
SOUTHEAST FLORIDA BEHAVIORAL HE	NEW HORIZONS OF THE TREASURE COAST	2.64	0.46	3.10	0.48	2.63
	Total	**2.64**	**0.46**	**3.10**	**0.48**	**2.63**
Grand Total		**2.64**	**0.46**	**3.10**	**0.48**	**2.63**

FIGURE 22 - INDIGENT CLIENTS SERVED REPORT OBSERVATIONS

Operational Beds, Purchased Beds, and Occupied Beds Report

The purpose of this report is to provide, in a tabular form, the average daily census of operational beds, the number of these beds purchased by DCF, and the number of operational beds that are occupied and billable per payor class, including DCF, Medicaid, Medicare, Private Insurance, Local match, and other payor classes.

Details of the three runs:

- First run: All options selected for multiselect fields with the biggest time frame possible selected. Response time (4 secs)
- Second run: All options selected for multiselect fields with the biggest time frame possible selected. Response time (4 secs)
- Third run: All options selected for multiselect fields with the biggest time frame possible selected. Response time (4 secs)

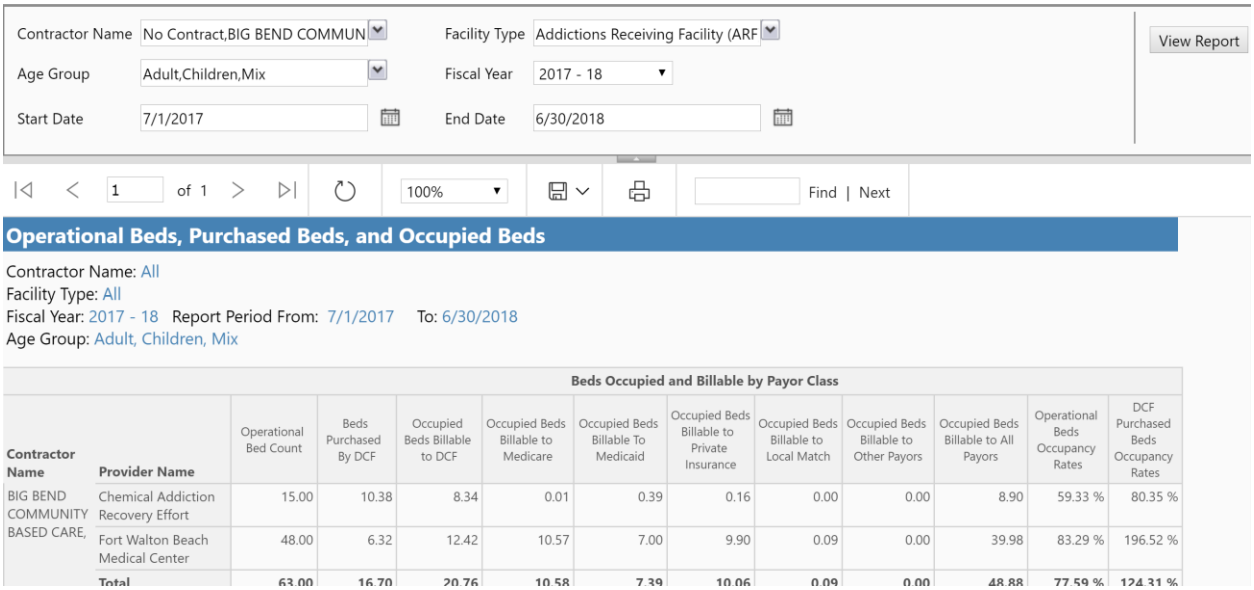


FIGURE 23 - OPERATIONAL BEDS, PURCHASED BEDS, AND OCCUPIED BEDS EXECUTION SAMPLE

Observations:

- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.
- There is no option to select multiple fiscal years.
- No validation for Start and End date (End date should be greater or equal than start date)
- Fiscal year does not filter Begin Date and End Date

Sample report:

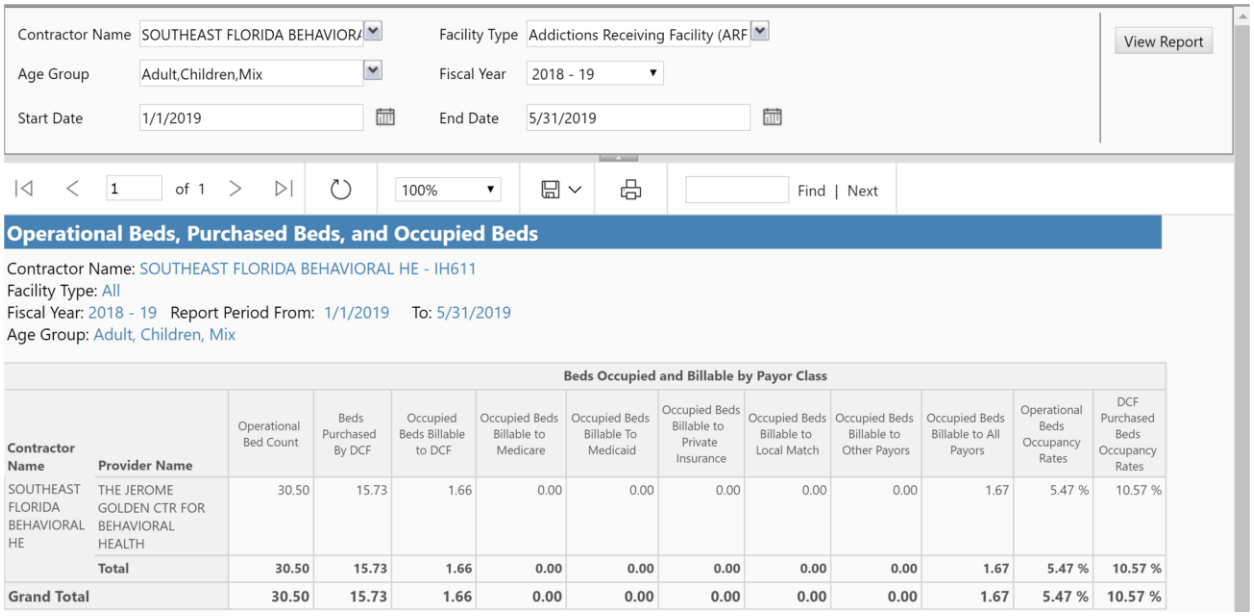


FIGURE 24 - OPERATIONAL BEDS, PURCHASED BEDS, AND OCCUPIED BEDS OBSERVATIONS

Operational Bed Occupancy Rates Report

The purpose of this report is to provide, in a tabular and graphical forms, the average daily census of total licensed beds, the number of these beds occupied by all payor classes, and the occupancy rates of these beds per month in each selected fiscal year.

Details of the three runs:

- First run: Default options selected, Response time (3 secs)
- Second Run: All options selected for multiselect fields, Response time (4 secs)
- Third Run: All options selected for multiselect fields, Response time (4 secs)

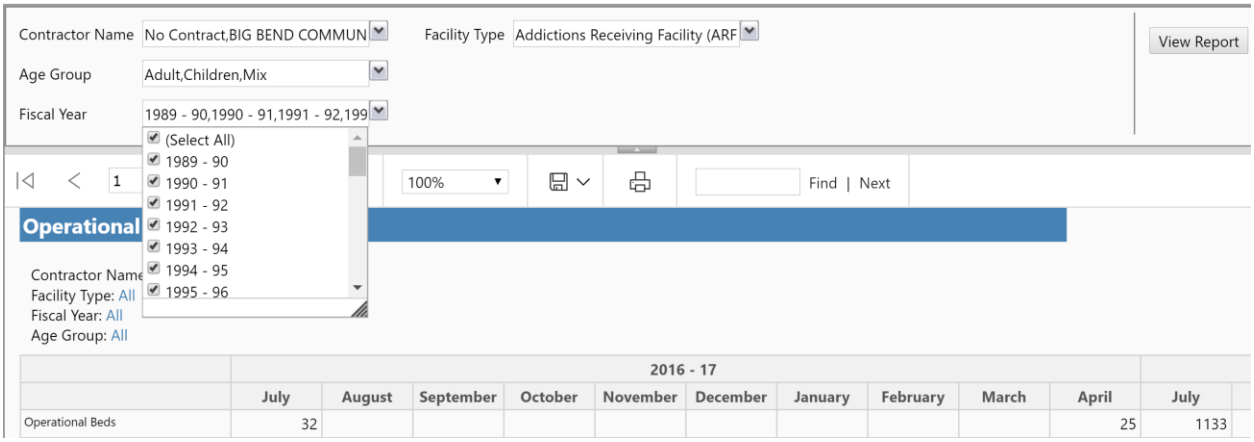


FIGURE 25 - OPERATIONAL BED OCCUPANCY RATES REPORT EXECUTION SAMPLE

Observations:

- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.
- All parameters have multiple selection available.
- Export functionality needs to use a better approach to present data. Files generated should present raw data instead.

Sample report:

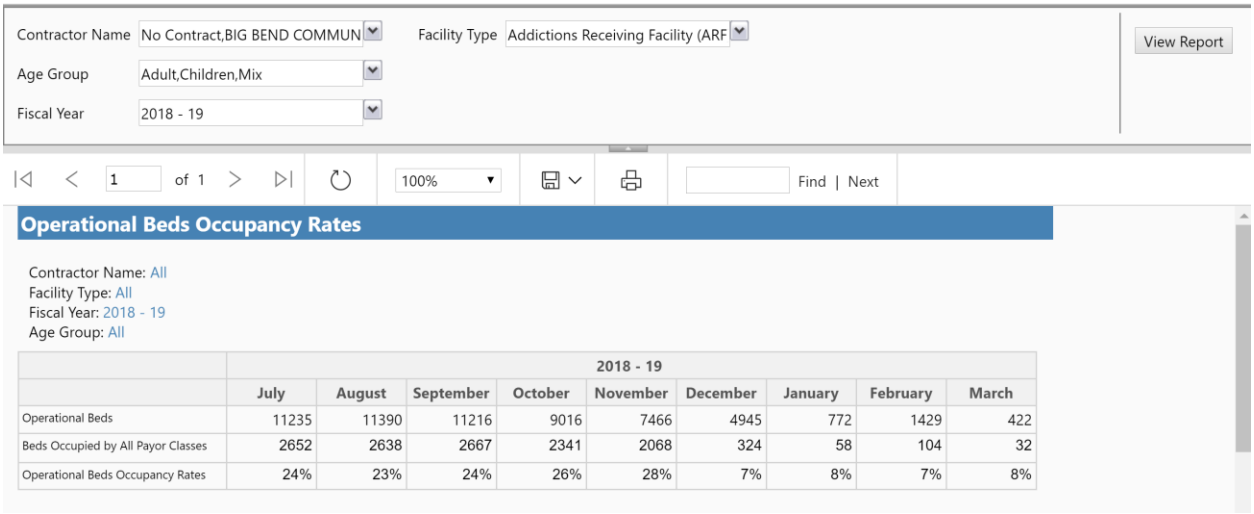


FIGURE 26 - OPERATIONAL BEDS OCCUPANCY RATES OBSERVATIONS (1 OF 2)

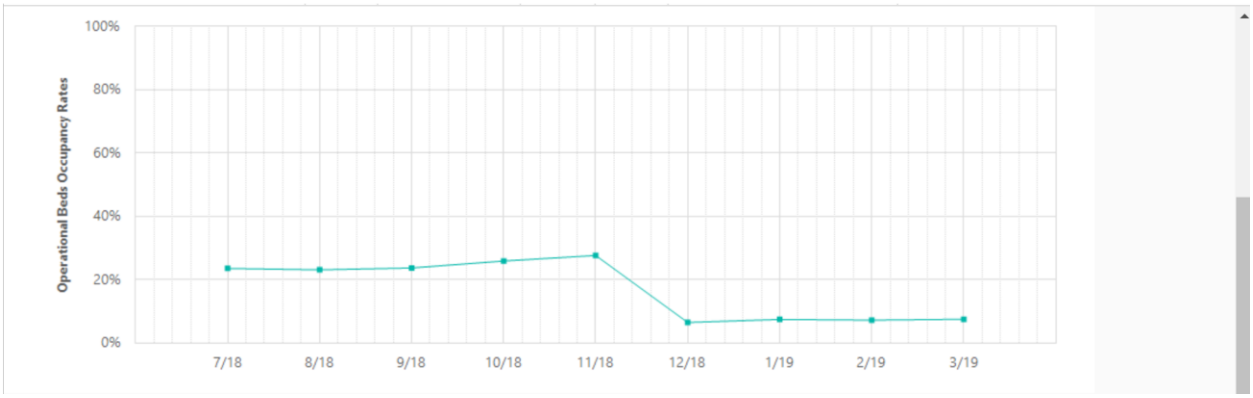


FIGURE 27 - OPERATIONAL BEDS OCCUPANCY RATES OBSERVATIONS (2 OF 2)

DCF Purchased Bed Occupancy Rates Report

The purpose of this report is to provide, in a tabular and graphical forms, the average daily census of total beds purchased by DCF, the number of these beds occupied by DCF payor class, and the occupancy rates of these beds per month in each fiscal year.

Details of the three runs:

- First run: All options selected for multiselect fields (except fiscal year) using default values. Response time (4 secs)
- Second run: All options selected for multiselect fields. Response time (5 secs)
- Third run: All options selected for multiselect fields. Response time (5 secs)

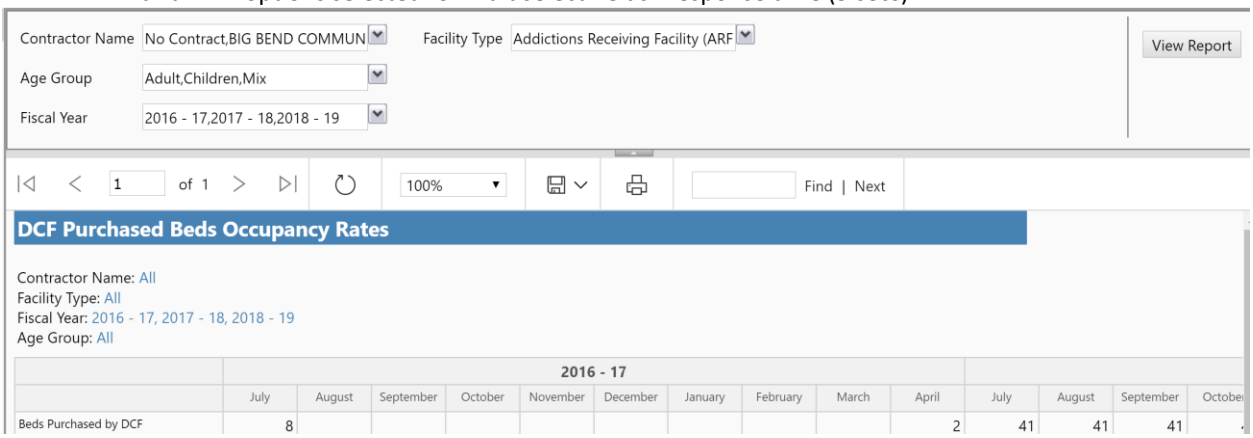
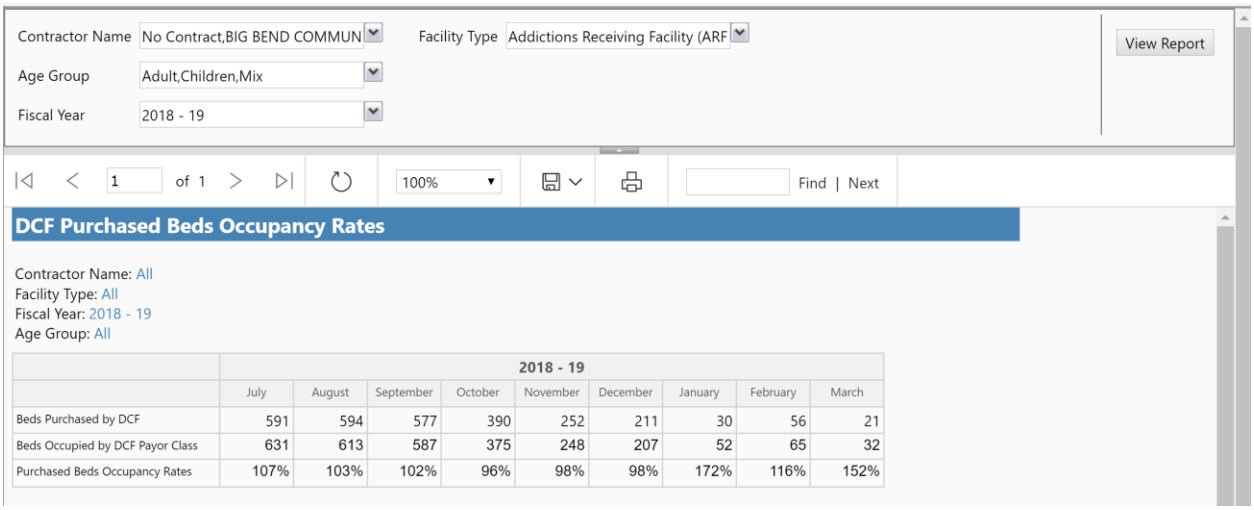


FIGURE 28 - DCF PURCHASED BEDS OCCUPANCY RATES EXECUTION SAMPLE

Observations:

- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.

Sample report:



	2018 - 19								
	July	August	September	October	November	December	January	February	March
Beds Purchased by DCF	591	594	577	390	252	211	30	56	21
Beds Occupied by DCF Payor Class	631	613	587	375	248	207	52	65	32
Purchased Beds Occupancy Rates	107%	103%	102%	96%	98%	98%	172%	116%	152%

FIGURE 29 - DCF PURCHASED BEDS OCCUPANCY RATES OBSERVATIONS (1 OF 2)

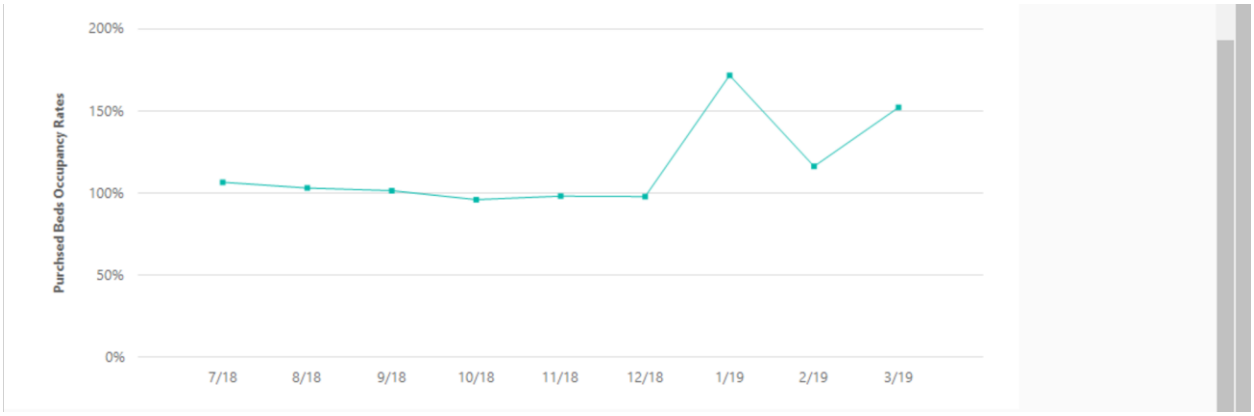


FIGURE 30 - DCF PURCHASED BEDS OCCUPANCY RATES OBSERVATIONS (2 OF 2)

Total Licensed Beds Report

This report provides, in a tabular form, the total number of licensed beds by contractor, provider, data source, license number, facility type and age group.

Details of the three runs:

- First run: All options selected for multiselect fields (except fiscal year) using default values. Response time (4 secs)
- Second run: All options selected for multiselect fields using default values. Response time (5 secs)
- Third run: All options selected for multiselect fields using default values. Response time (5 secs)

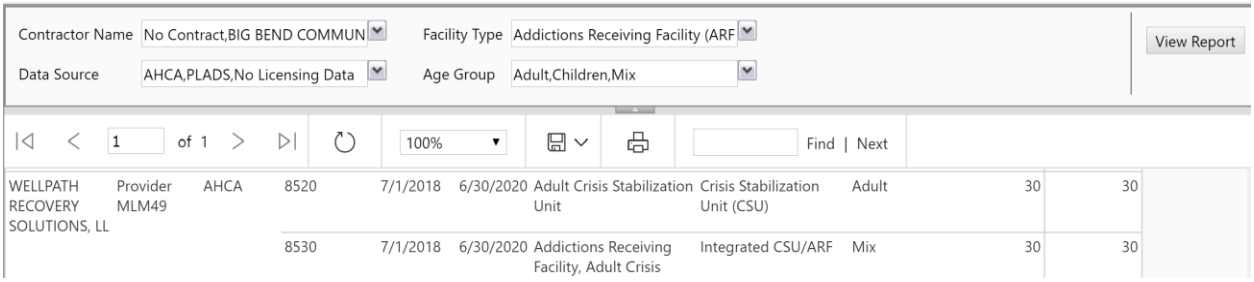


FIGURE 31 - TOTAL LICENSED BEDS REPORT EXECUTION SAMPLE

Observations:

- Provider is not an option in the report's parameters.
- Headings always need to be visible when scrolling down through the report details.

Sample report:

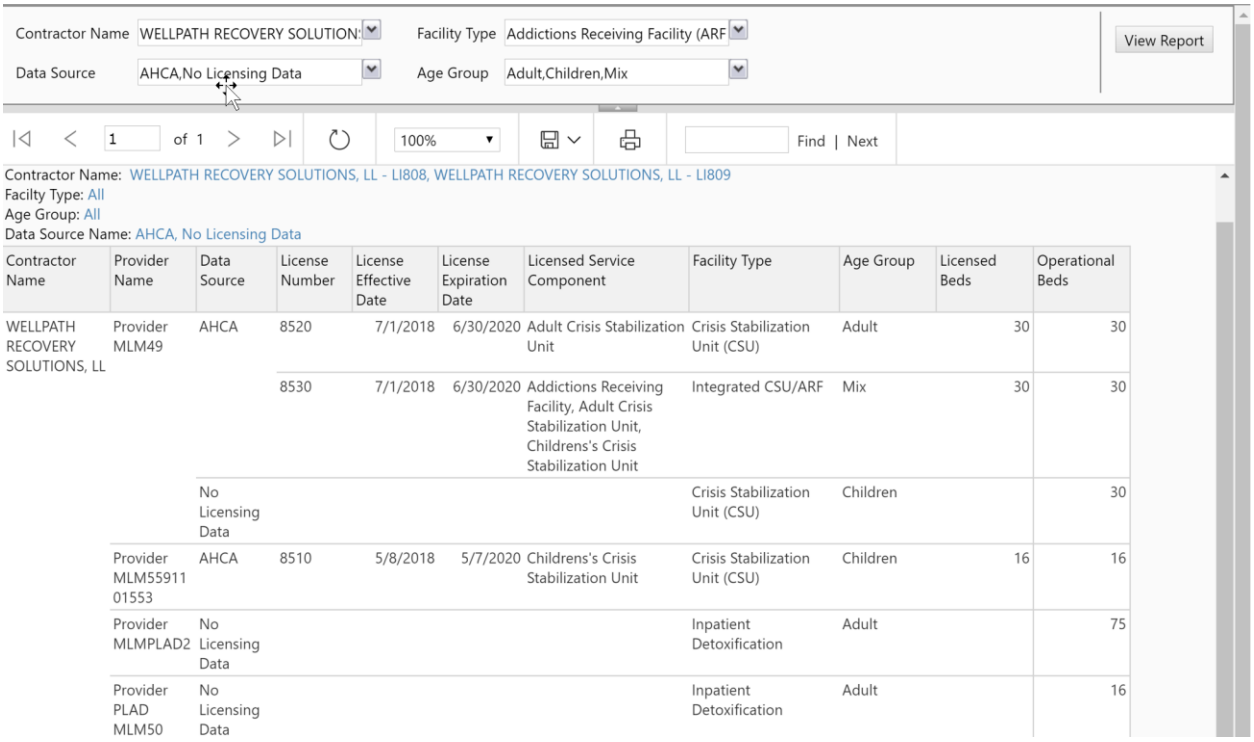


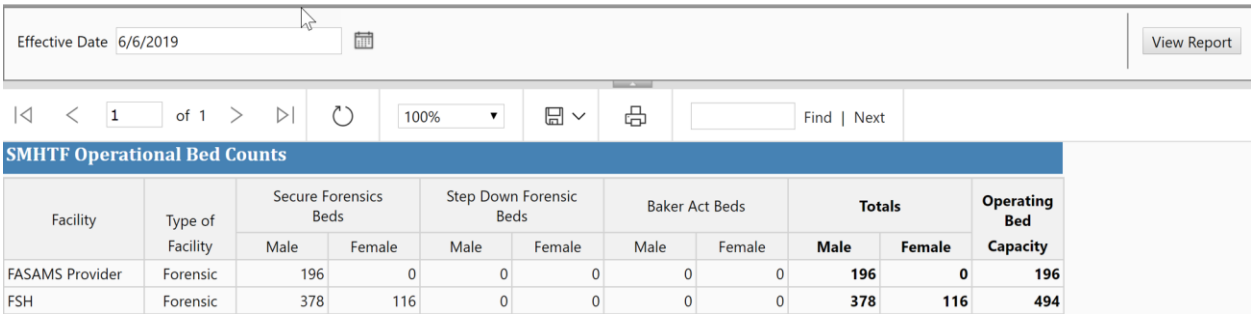
FIGURE 32 - TOTAL LICENSED BEDS REPORT EXECUTION OBSERVATIONS

SMHTF Operational Bed Counts Report

[No descriptive information was found by the team in the documentation to provide a report description]

Details of the three runs:

- First run: Default values. Response time (2 secs)
- Second run: Date in the past selected. Response time (2 secs)
- Third run: Date in the past selected. Response time (2 secs)



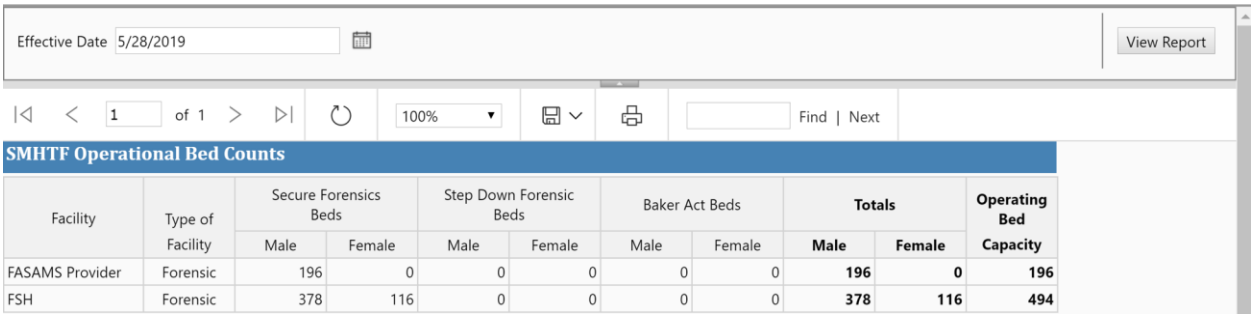
Facility	Type of Facility	Secure Forensics Beds		Step Down Forensic Beds		Baker Act Beds		Totals		Operating Bed Capacity
		Male	Female	Male	Female	Male	Female	Male	Female	
FASAMS Provider	Forensic	196	0	0	0	0	0	**196**	**0**	**196**
FSH	Forensic	378	116	0	0	0	0	**378**	**116**	**494**

FIGURE 33 - SMHTF OPERATIONAL BED COUNTS REPORT EXECUTION SAMPLE

Observations:

- There is no way to select the Facility in the report's parameters.
- Headings always need to be visible when scrolling down through the report details.

Sample report:



Facility	Type of Facility	Secure Forensics Beds		Step Down Forensic Beds		Baker Act Beds		Totals		Operating Bed Capacity
		Male	Female	Male	Female	Male	Female	Male	Female	
FASAMS Provider	Forensic	196	0	0	0	0	0	**196**	**0**	**196**
FSH	Forensic	378	116	0	0	0	0	**378**	**116**	**494**

FIGURE 34 - SMHTF OPERATIONAL BED COUNTS REPORT OBSERVATIONS

Client Reports

[No descriptive information was found by the team in the documentation to provide a report description]

Client Search Report

The report will allow submitting entity users to search for clients by a variety of fields, as a method for obtaining the unique client identifier returned by the MCI system for an individual client. DCF staff may also run this report to assist submitting entities with verifying client information.

Details of the three runs:

- First run: All multiselect options. Last name entered (a*). Response time (7 seconds)
- Second run: All multiselect options. Last name entered (smith). Response time (7 seconds)
- Third run: All multiselect options. Last name entered (b*). Response time (8 seconds)

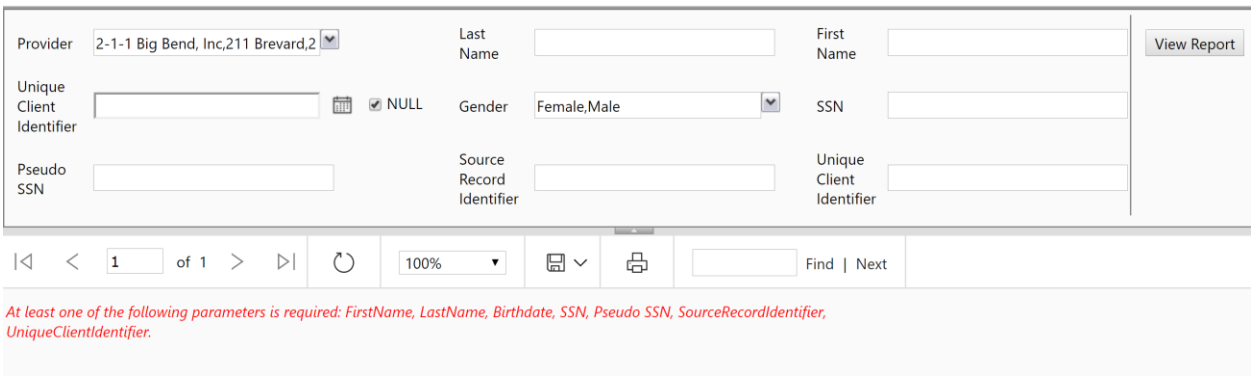
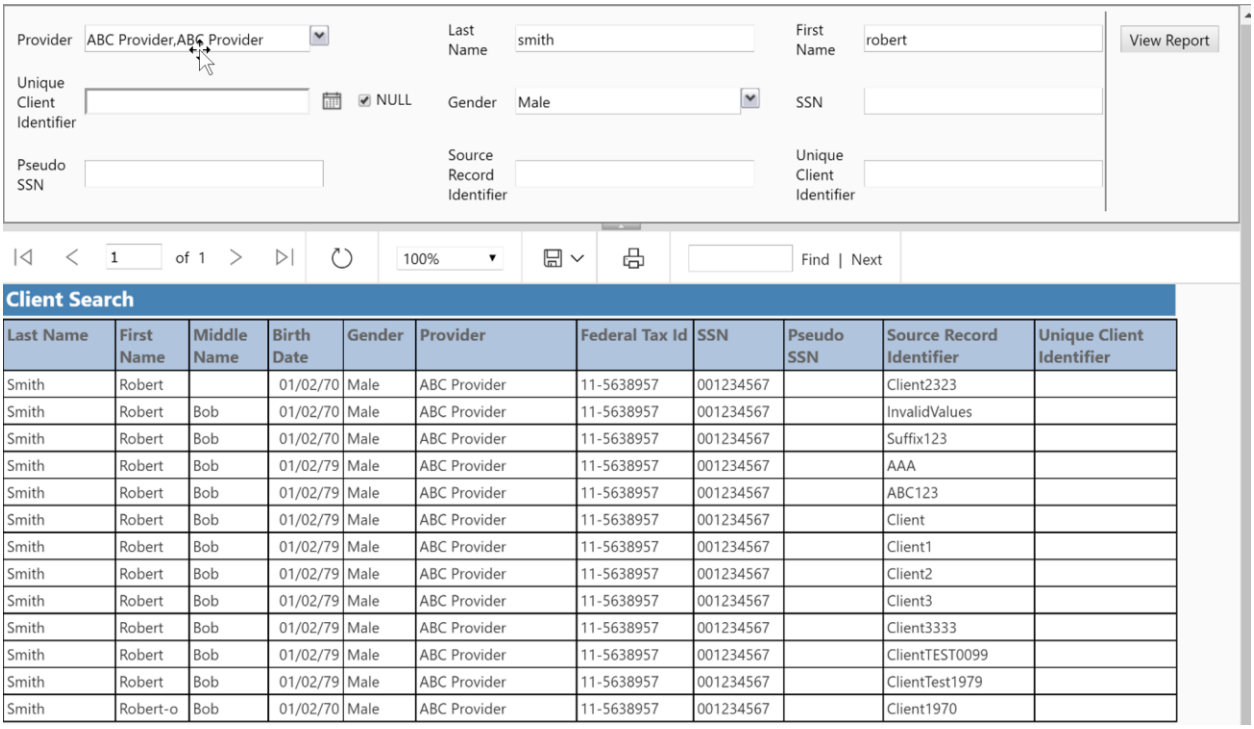


FIGURE 35 - CLIENT SEARCH REPORT EXECUTION SAMPLE

Observations:

- Headings always need to be visible when scrolling down through the report details.
- Misleading validation message (Birthdate mandatory and there is not birthdate parameter to enter).

Sample report:



Client Search

Last Name	First Name	Middle Name	Birth Date	Gender	Provider	Federal Tax Id	SSN	Pseudo SSN	Source Record Identifier	Unique Client Identifier
Smith	Robert		01/02/70	Male	ABC Provider	11-5638957	001234567		Client2323	
Smith	Robert	Bob	01/02/70	Male	ABC Provider	11-5638957	001234567		InvalidValues	
Smith	Robert	Bob	01/02/70	Male	ABC Provider	11-5638957	001234567		Suffix123	
Smith	Robert	Bob	01/02/79	Male	ABC Provider	11-5638957	001234567		AAA	
Smith	Robert	Bob	01/02/79	Male	ABC Provider	11-5638957	001234567		ABC123	
Smith	Robert	Bob	01/02/79	Male	ABC Provider	11-5638957	001234567		Client	
Smith	Robert	Bob	01/02/79	Male	ABC Provider	11-5638957	001234567		Client1	
Smith	Robert	Bob	01/02/79	Male	ABC Provider	11-5638957	001234567		Client2	
Smith	Robert	Bob	01/02/79	Male	ABC Provider	11-5638957	001234567		Client3	
Smith	Robert	Bob	01/02/79	Male	ABC Provider	11-5638957	001234567		Client3333	
Smith	Robert	Bob	01/02/79	Male	ABC Provider	11-5638957	001234567		ClientTEST0099	
Smith	Robert	Bob	01/02/79	Male	ABC Provider	11-5638957	001234567		ClientTest1979	
Smith	Robert-o	Bob	01/02/70	Male	ABC Provider	11-5638957	001234567		Client1970	

FIGURE 36 - CLIENT SEARCH REPORT OBSERVATIONS

Contract Compliance Reports

[No descriptive information was found by the team in the documentation to provide a report description]

Output Measures – Cumulative Report

The purpose of this report is to calculate the monthly accumulated number of clients served in each service category by the Program Areas: Adult Mental Health, Children's Mental Health, Adult Substance Abuse and Children's

Substance Abuse for a selected Fiscal Year. This report will provide the flexibility to view the total number of clients served at five different hierarchies:

- State
- Contractor
- Provider-Subcontract
- OCA and Covered Service
- Client

Also, users will be able to filter the report by Contracted Entities, Providers and Provider Sites.

Details of the three runs:

- First run: All options selected for multiselect fields (except fiscal year) using default values. Response time (7 seconds)
- Second run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)
- Third run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)

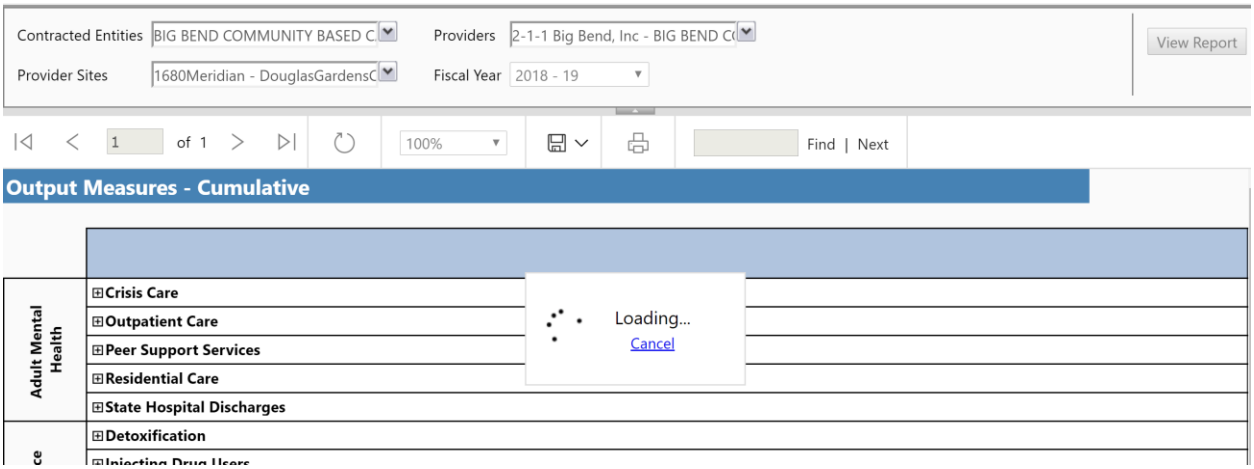


FIGURE 37 - OUTPUT MEASURES - CUMULATIVE REPORT EXECUTION SAMPLE

Observations:

- The following message is displayed when export the report to excel is performed and then opened in a spreadsheet.

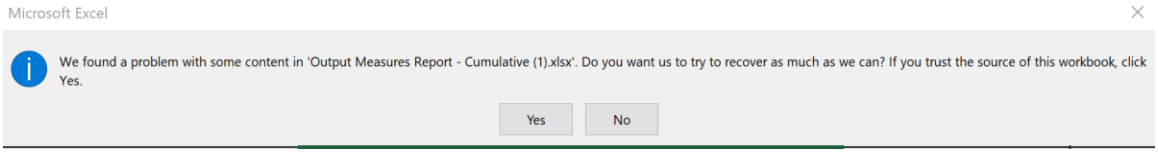


FIGURE 38 - OUTPUT MEASURES - CUMULATIVE REPORT EXPORT ERROR

- Headings always need to be visible when scrolling down through the report details.
- Some dropdowns have too many options and there is no filtering/search feature in place.
- Only a single year can be selected. Recommendation: Select multiple years capabilities.
- Grouping of data helps for visualization but expanding/collapsing groups take additional 7 seconds
- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.

Sample report:

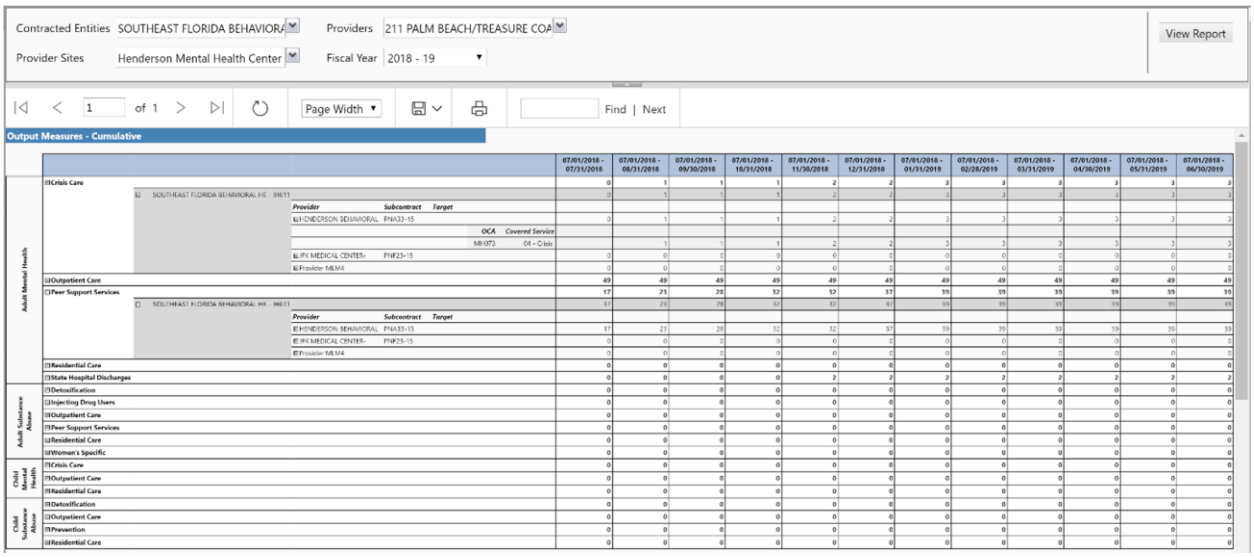


FIGURE 39 - OUTPUT MEASURES - CUMULATIVE REPORT OBSERVATIONS

Output Measures – Monthly Report

The purpose of this report is to calculate the number of clients served in each service category by the Program Areas: Adult Mental Health, Children's Mental Health, Adult Substance Abuse and Children's Substance Abuse for a selected Fiscal Year within each individual month.

This report is very similar to Output Measures Report – Cumulative, but each column will calculate the clients served within each individual month.

Details of the three runs:

- First run: All options selected for multiselect fields (except fiscal year) using default values. Response time (7 seconds)
- Second run: All options selected for multiselect fields using previous fiscal year. Response time fair (7 seconds)
- Third run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)

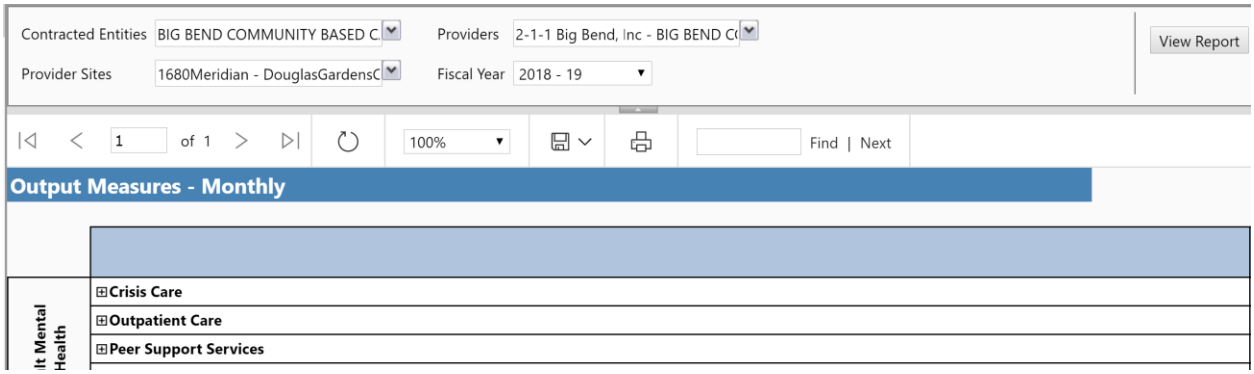


FIGURE 40 - OUTPUT MEASURES MONTHLY REPORT EXECUTION SAMPLE

Observations:

- Grouping of data helps for visualization but expanding/collapsing groups take additional 7 seconds
- Headings always need to be visible when scrolling down through the report details.
- Some dropdowns have too many options and there is no filtering/search feature in place.
- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.

Sample report:

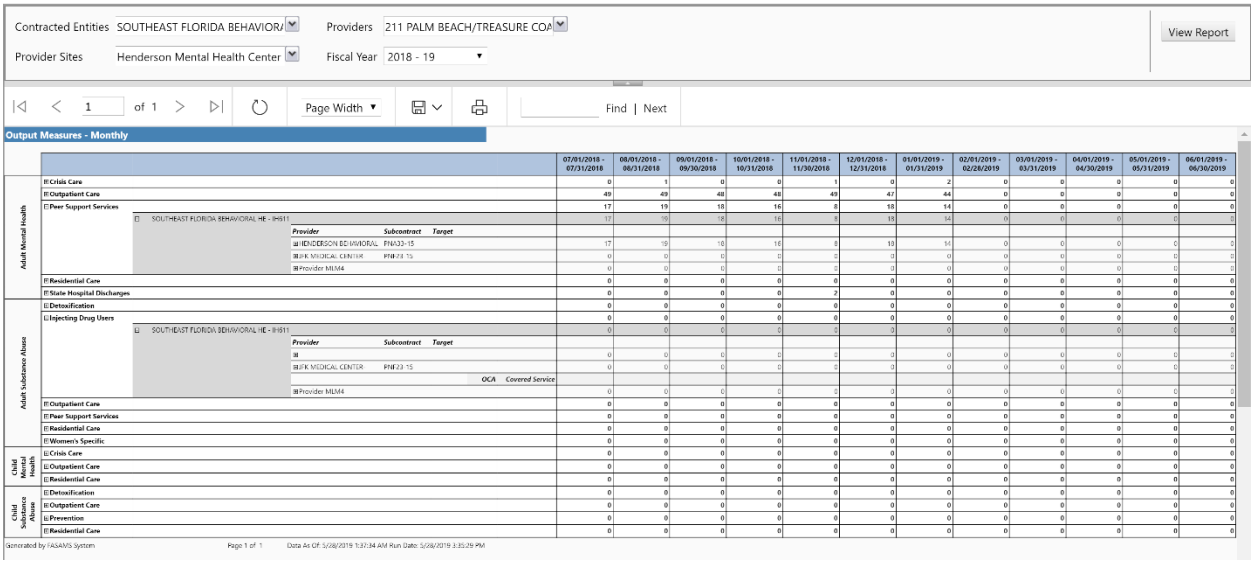


FIGURE 41 - OUTPUT MEASURES MONTHLY REPORT OBSERVATIONS

Outcome Measures – Cumulative Performance Report

The purpose of this report is to provide the actual performance data in order to access the level of contract compliance throughout the state in a fiscal year. This report will contain two tables:

- Outcome Measures table will calculate the monthly accumulated outcome measures in four Program Areas: Adult Mental Health, Children's Mental Health, Adult Substance Abuse and Children's Substance Abuse.
- Performance Measure table will calculate the contract compliance level at the state, contract and subcontract levels based on the outcome measures results.

Both tables will provide the ability to view data at four different hierarchies:

- State
- Contractor
- Provider-Subcontract
- Client

Also, users will be able to filter the report by Contracted Entities, Providers and Provider Sites.

Details of the three runs:

- First run: All options selected for multiselect fields (except fiscal year) using default values. Response time (7 seconds)
- Second run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)
- Third run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)

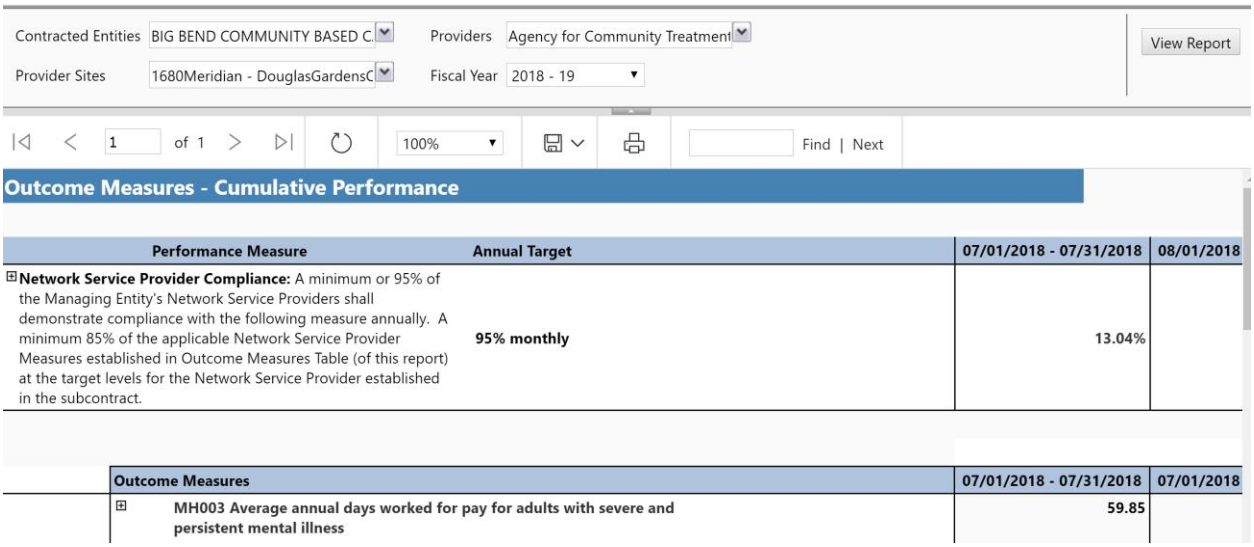


FIGURE 42 - OUTCOME MEASURES CUMULATIVE PERFORMANCE REPORT EXECUTION SAMPLE

Observations:

- Headings always need to be visible when scrolling down through the report details.
- Some dropdowns have too many options and there is no filtering/search feature in place.
- Only a single year can be selected. Recommendation: Select multiple years capabilities.
- Grouping of data helps for visualization but expanding/collapsing groups take additional 7 seconds
- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.

Sample report:

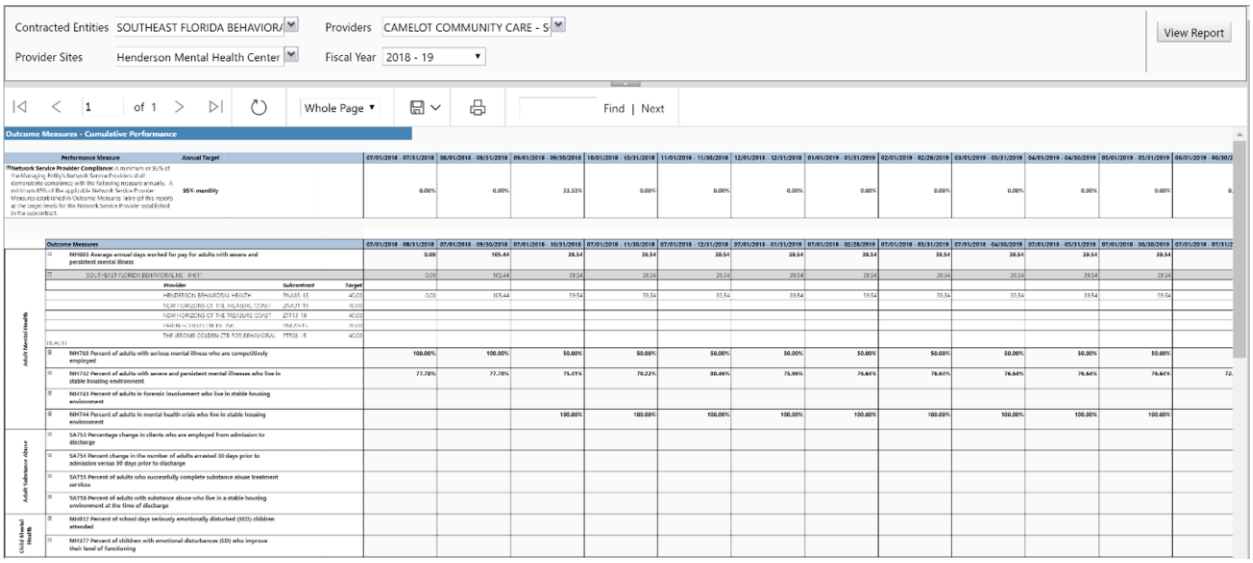


FIGURE 43 - OUTCOME MEASURES CUMULATIVE PERFORMANCE REPORT OBSERVATIONS

Outcome Measures – Cumulative Data Set Size Report

The purpose of this report is to provide the sizes of the data sets that the Outcome Measures table of Outcome Measures Report – Cumulative Performance is based on. Therefore, this report will be very similar to Outcome

Measures Report – Cumulative Performance (Section **Error! Reference source not found.**), but contains only the O utcome Measures table that reports data set sizes instead of performance.

Details of the three runs:

- First run: All options selected for multiselect fields (except fiscal year) using default values. Response time (7 seconds)
- Second run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)
- Third run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)

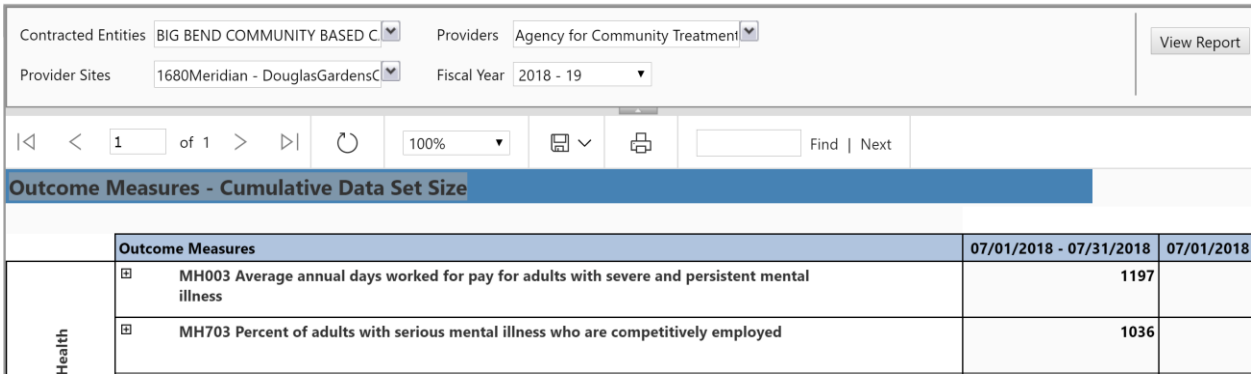


FIGURE 44 - OUTCOME MEASURES CUMULATIVE DATA SET SIZE REPORT EXECUTION SAMPLE

Observations:

- Headings always need to be visible when scrolling down through the report details.
- Some dropdowns have too many options and there is no filtering/search feature in place.
- Only a single year can be selected. Recommendation: Select multiple years capabilities.
- Grouping of data helps for visualization but expanding/collapsing groups take additional 7 seconds
- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.

Sample report:

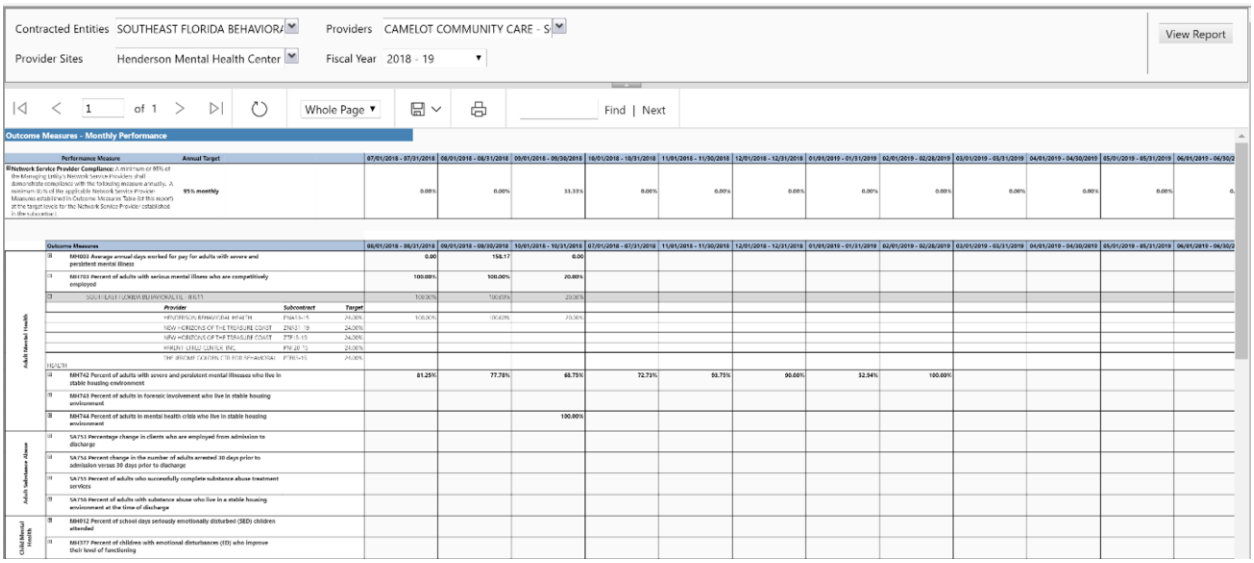


FIGURE 45 - OUTCOME MEASURES CUMULATIVE DATA SET SIZE REPORT OBSERVATIONS (1 OF 2)

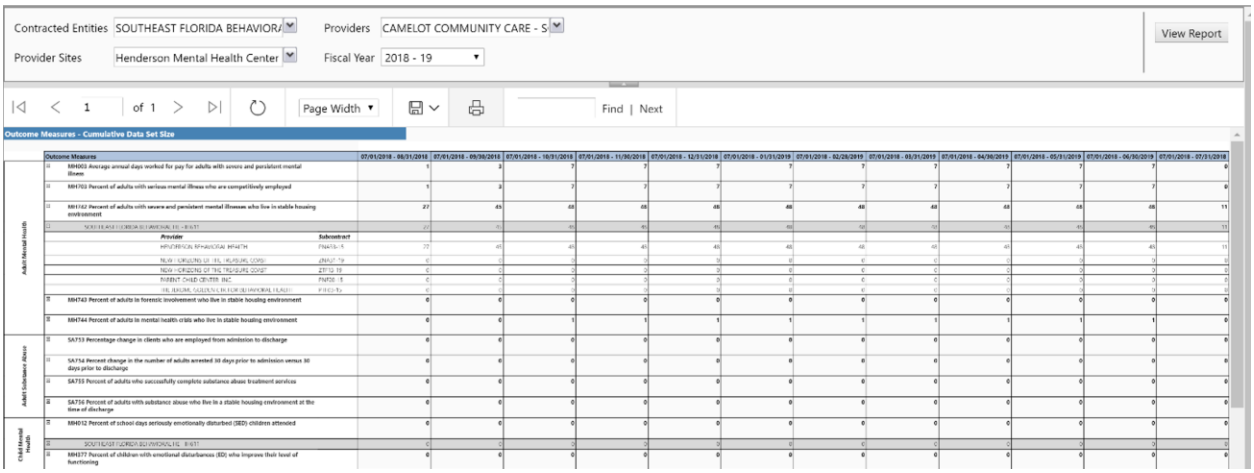


FIGURE 46 - OUTCOME MEASURES CUMULATIVE DATA SET SIZE REPORT OBSERVATIONS (2 OF 2)

Outcome Measures – Monthly Performance Report

The purpose of this report is to provide the actual performance data in order to access the level of contract compliance throughout the state in a fiscal year. This report will contain two tables:

- Outcome Measures table will calculate each month's outcome measures in four Program Areas: Adult Mental Health, Children's Mental Health, Adult Substance Abuse and Children's Substance Abuse.
- Performance Measure table will calculate the contract compliance level at the state, contract and subcontract levels based on the outcome measures results.

This report is very similar to Outcome Measures Report – Cumulative Performance (Section **Error! Reference source not found.**), but each column will calculate the actual performance within each individual month.

Details of the three runs:

- First run: All options selected for multiselect fields (except fiscal year) using default values. Response time (7 seconds)
- Second run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)
- Third run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)

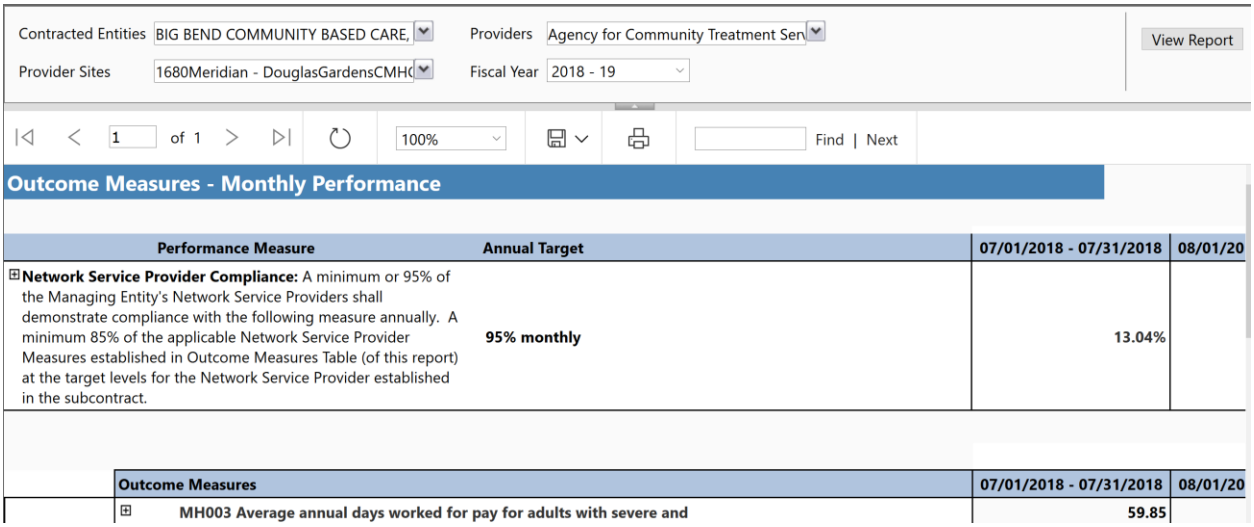


FIGURE 47 - OUTCOME MEASURES MONTHLY PERFORMANCE REPORT EXECUTION SAMPLE

Observations:

- Headings always need to be visible when scrolling down through the report details.
- Some dropdowns have too many options and there is no filtering/search feature in place.
- Only a single year can be selected. Recommendation: Select multiple years capabilities.
- Grouping of data helps for visualization but expanding/collapsing groups take additional 7 seconds
- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.

Sample report:

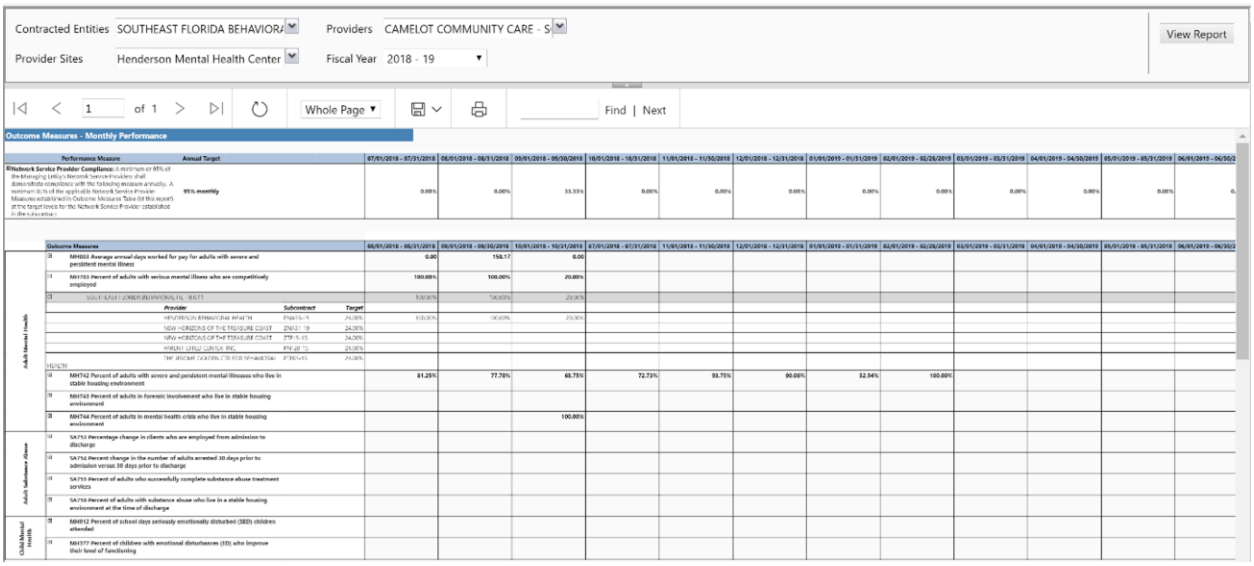


FIGURE 48 - OUTCOME MEASURES MONTHLY PERFORMANCE REPORT OBSERVATIONS

Outcome Measures – Monthly Data Set Size Report

The purpose of this report is to provide the sizes of the data sets that the Outcome Measures table of Outcome Measures Report – Monthly Performance is based on. Therefore, this report will be very similar to Outcome Measures Report – Monthly Performance (Section **Error! Reference source not found.**), but contains only the O utcome Measures table that reports data set sizes instead of performance.

Details of the three runs:

- First run: All options selected for multiselect fields (except fiscal year) using default values. Response time (7 seconds)
- Second run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)
- Third run: All options selected for multiselect fields using previous fiscal year. Response time (7 seconds)

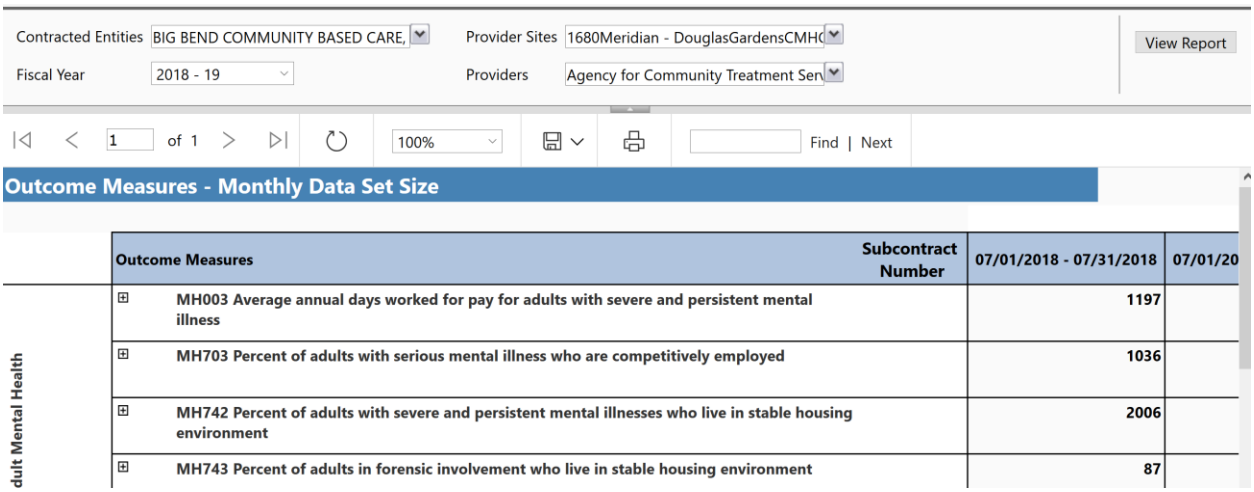


FIGURE 49 - OUTCOME MEASURES - MONTHLY DATA SET SIZE REPORT EXECUTION SAMPLE

Observations:

- Headings always need to be visible when scrolling down through the report details.
- Some dropdowns have too many options and there is no filtering/search feature in place.
- Only a single year can be selected. Recommendation: Select multiple years capabilities.
- Grouping of data helps for visualization but expanding/collapsing groups take additional 7 seconds
- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data.

Sample report:

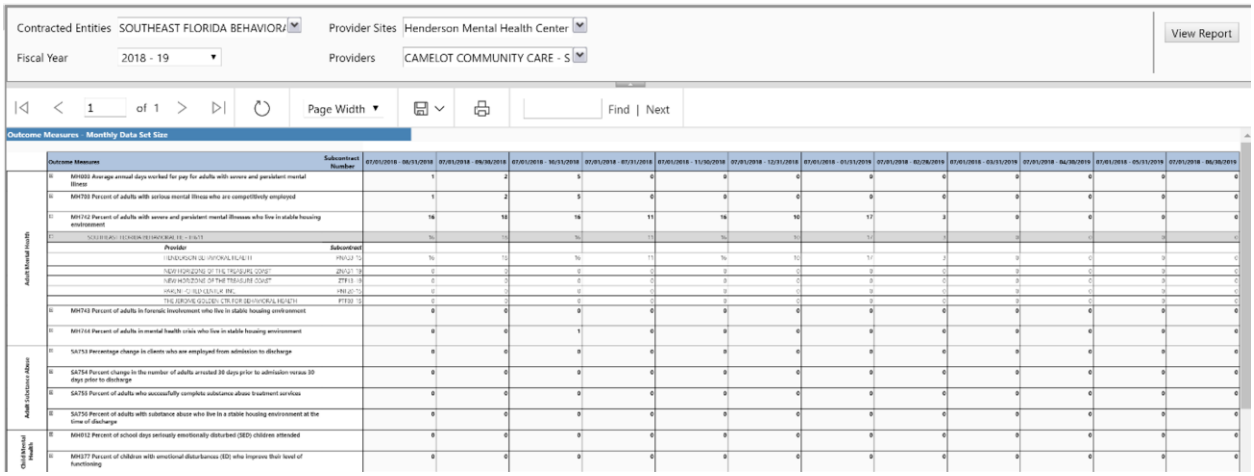


FIGURE 50 - OUTCOME MEASURES - MONTHLY DATA SET SIZE REPORT OBSERVATIONS

Exception Reports

[No descriptive information was found by the team in the documentation to provide a report description]

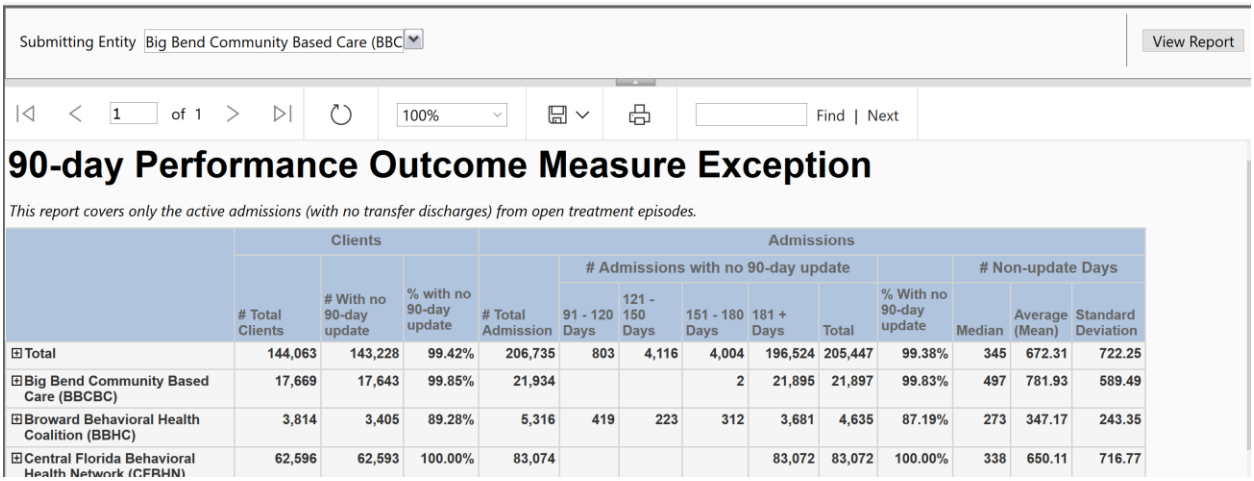
90 Day Performance Outcome Measure Exception Report

FASAMS Pamphlet 155-2 Chapter 5 Treatment Episode Data requires submitting entities to submit Performance Outcome Measurements updated every 90 days following the initial admission or transfer admission and the individual does not have a Final Discharge for the Treatment Episode. If a submitting entity does not complete a

Performance Outcome Measure within 90 days of the last Performance Outcome Measurements, it will be included in an exception report.

Details of the three runs:

- First run: All options selected for multiselect fields using default values. Response time is (5 secs)
- Second run: Selecting three values from the only dropdown available. Response time is (4 secs)
- Third run: Selecting three values from the only dropdown available. Response time is (4 secs)



	Clients			Admissions								
					# Admissions with no 90-day update					# Non-update Days		
	# Total Clients	# With no 90-day update	% with no 90-day update	# Total Admission	91 - 120 Days	121 - 150 Days	151 - 180 Days	181 + Days	Total	% With no 90-day update	Median	Average (Mean)	Standard Deviation
⊞Total	144,063	143,228	99.42%	206,735	803	4,116	4,004	196,524	205,447	99.38%	345	672.31	722.25
⊞Big Bend Community Based Care (BBCBC)	17,669	17,643	99.85%	21,934			2	21,895	21,897	99.83%	497	781.93	589.49
⊞Broward Behavioral Health Coalition (BBHC)	3,814	3,405	89.28%	5,316	419	223	312	3,681	4,635	87.19%	273	347.17	243.35
⊞Central Florida Behavioral Health Network (CFBHN)	62,596	62,593	100.00%	83,074				83,072	83,072	100.00%	338	650.11	716.77

FIGURE 51 - 90 DAY PERFORMANCE OUTCOME MEASURE EXCEPTION REPORT EXECUTION SAMPLE

Observations:

- Headings always need to be visible when scrolling down through the report details.
- Some dropdowns have too many options and there is no filtering/search feature in place.
- Grouping of data helps for visualization but expanding/collapsing groups takes additional 7 seconds – longer than the report takes to run in the first place

Sample report:

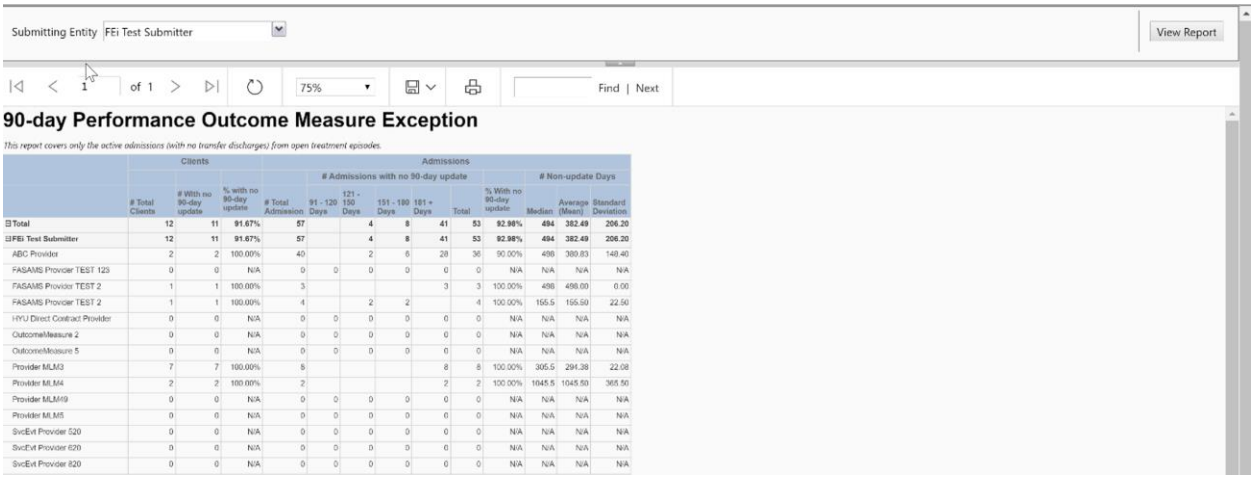


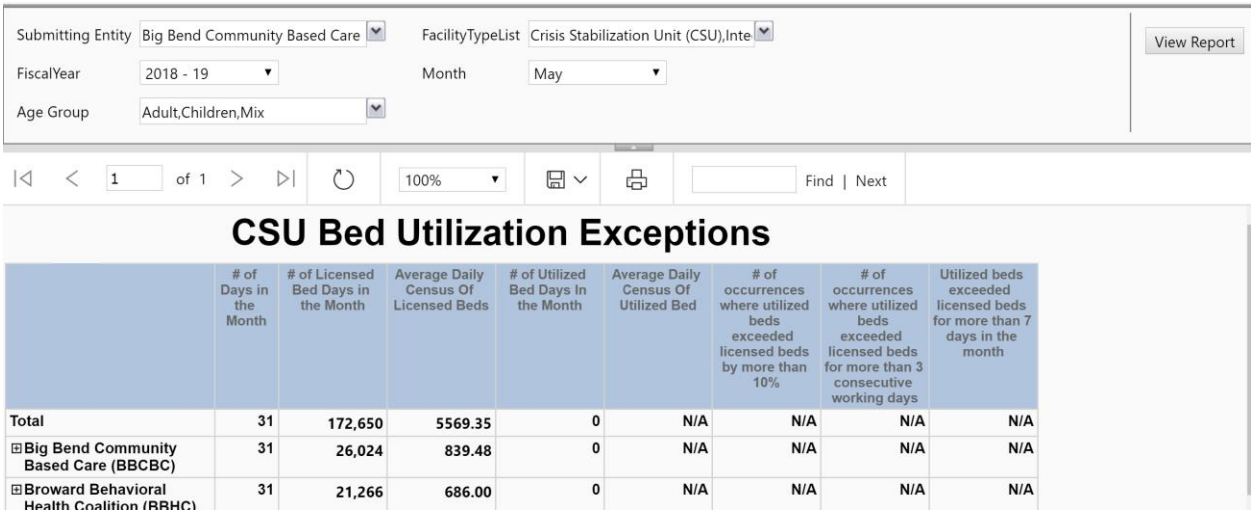
FIGURE 52 - 90 DAY PERFORMANCE OUTCOME MEASURE EXCEPTION REPORT OBSERVATIONS

CSU Bed Utilization Exception Report

[No descriptive information was found by the team in the documentation to provide a report description]

Details of the three runs:

- First run: All options selected for multiselect fields (except for fiscal year and month) using default values. Response time (4 secs)
- Second run: All options selected for multiselect fields (except for fiscal year and month) using a previous fiscal year. Response time (4 secs)
- Third run: All options selected for multiselect fields (except for fiscal year and month) using a previous fiscal year. Response time (4 secs)



CSU Bed Utilization Exceptions

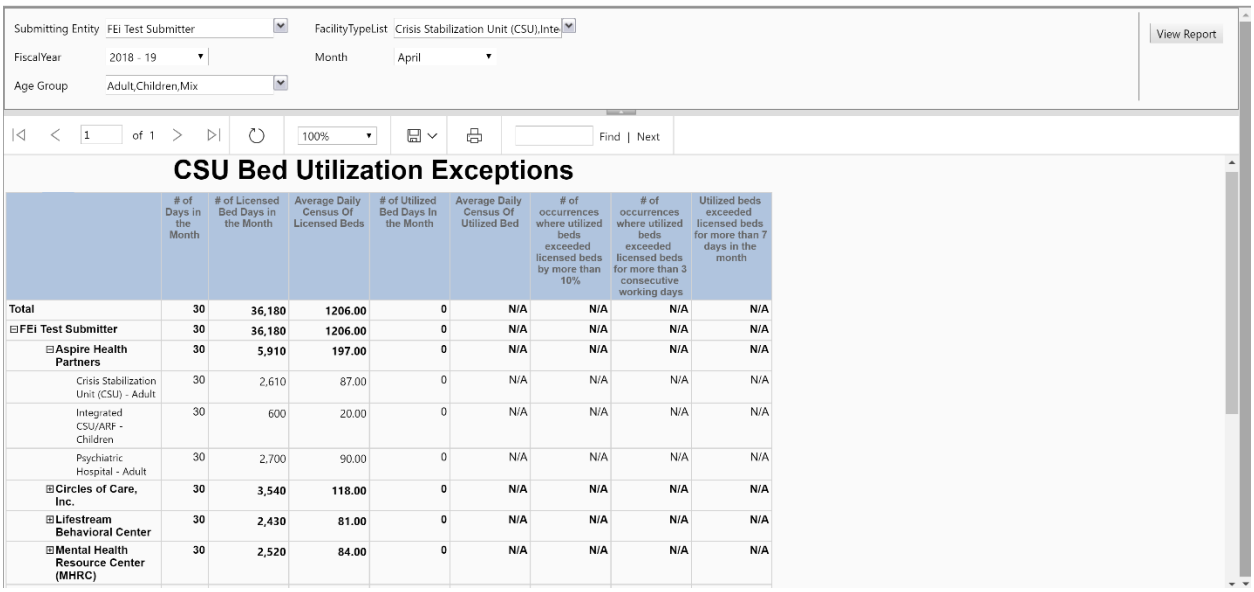
	# of Days in the Month	# of Licensed Bed Days in the Month	Average Daily Census Of Licensed Beds	# of Utilized Bed Days In the Month	Average Daily Census Of Utilized Bed	# of occurrences where utilized beds exceeded licensed beds by more than 10%	# of occurrences where utilized beds exceeded licensed beds for more than 3 consecutive working days	Utilized beds exceeded licensed beds for more than 7 days in the month
Total	31	172,650	5569.35	0	N/A	N/A	N/A	N/A
⊞Big Bend Community Based Care (BBCBC)	31	26,024	839.48	0	N/A	N/A	N/A	N/A
⊞Broward Behavioral Health Coalition (BBHC)	31	21,266	686.00	0	N/A	N/A	N/A	N/A

FIGURE 53 - CSU BED UTILIZATION EXCEPTION REPORT EXECUTION SAMPLE

Observations:

- Headings always need to be visible when scrolling down through the report details.
- Some dropdowns have too many options and there is no filtering/search feature in place.
- Only a single year can be selected. Recommendation: Select multiple years capabilities.
- Month (No multiple selection)
- Grouping of data helps for visualization but expanding/collapsing groups take additional 7 seconds
- Unnecessary options in Fiscal Year dropdown – there are date options for years where there is no available data

Sample report:



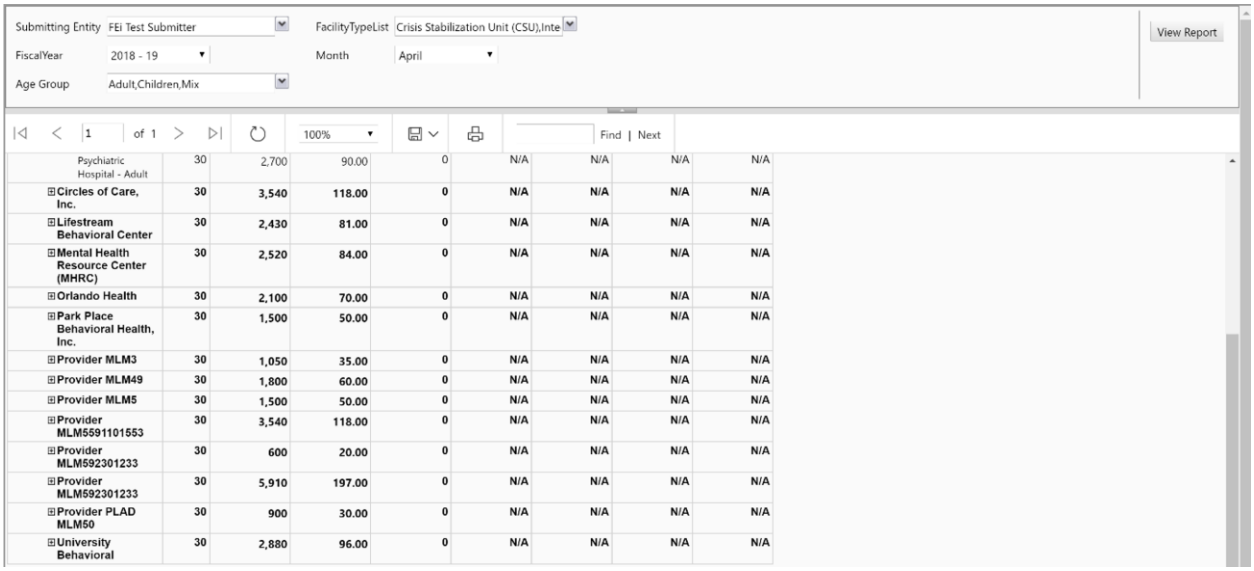
Submitting Entity: FEi Test Submitter | FacilityTypeList: Crisis Stabilization Unit (CSU),Inte | View Report
FiscalYear: 2018 - 19 | Month: April
Age Group: Adult,Children,Mix

1 of 1 | 100% | Find | Next

CSU Bed Utilization Exceptions

	# of Days in the Month	# of Licensed Bed Days in the Month	Average Daily Census Of Licensed Beds	# of Utilized Bed Days In the Month	Average Daily Census Of Utilized Bed	# of occurrences where utilized beds exceeded licensed beds by more than 10%	# of occurrences where utilized beds exceeded licensed beds for more than 3 consecutive working days	Utilized beds exceeded licensed beds for more than 7 days in the month
Total	**30**	**36,180**	**1206.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊟FEi Test Submitter	**30**	**36,180**	**1206.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊟Aspire Health Partners	**30**	**5,910**	**197.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
Crisis Stabilization Unit (CSU) - Adult	30	2,610	87.00	0	N/A	N/A	N/A	N/A
Integrated CSU/ARF - Children	30	600	20.00	0	N/A	N/A	N/A	N/A
Psychiatric Hospital - Adult	30	2,700	90.00	0	N/A	N/A	N/A	N/A
⊞Circles of Care, Inc.	**30**	**3,540**	**118.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Lifestream Behavioral Center	**30**	**2,430**	**81.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Mental Health Resource Center (MHRC)	**30**	**2,520**	**84.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**

FIGURE 54 - CSU BED UTILIZATION EXCEPTION REPORT OBSERVATIONS (1 OF 2)



Submitting Entity: FEi Test Submitter | FacilityTypeList: Crisis Stabilization Unit (CSU),Inte | View Report
FiscalYear: 2018 - 19 | Month: April
Age Group: Adult,Children,Mix

1 of 1 | 100% | Find | Next

Psychiatric Hospital - Adult	30	2,700	90.00	0	N/A	N/A	N/A	N/A
⊞Circles of Care, Inc.	**30**	**3,540**	**118.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Lifestream Behavioral Center	**30**	**2,430**	**81.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Mental Health Resource Center (MHRC)	**30**	**2,520**	**84.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Orlando Health	**30**	**2,100**	**70.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Park Place Behavioral Health, Inc.	**30**	**1,500**	**50.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Provider MLM3	**30**	**1,050**	**35.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Provider MLM49	**30**	**1,800**	**60.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Provider MLM5	**30**	**1,500**	**50.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Provider MLM5591101553	**30**	**3,540**	**118.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Provider MLM592301233	**30**	**600**	**20.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Provider MLM592301233	**30**	**5,910**	**197.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞Provider PLAD MLM50	**30**	**900**	**30.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**
⊞University Behavioral	**30**	**2,880**	**96.00**	**0**	**N/A**	**N/A**	**N/A**	**N/A**

FIGURE 55 - CSU BED UTILIZATION EXCEPTION REPORT OBSERVATIONS (2 OF 2)

Federal Reports

[No descriptive information was found by the team in the documentation to provide a report description]

SABG 7 – SABG Statewide Entity Inventory Report

This report provides a list of sub-recipients of SABG funds, which are the providers who provided substance abuse treatment services during the block grant expenditure period (two federal fiscal years). The following columns from the original SABG Table 7 will not be included in this report, because the required information is not available in FASAMS:

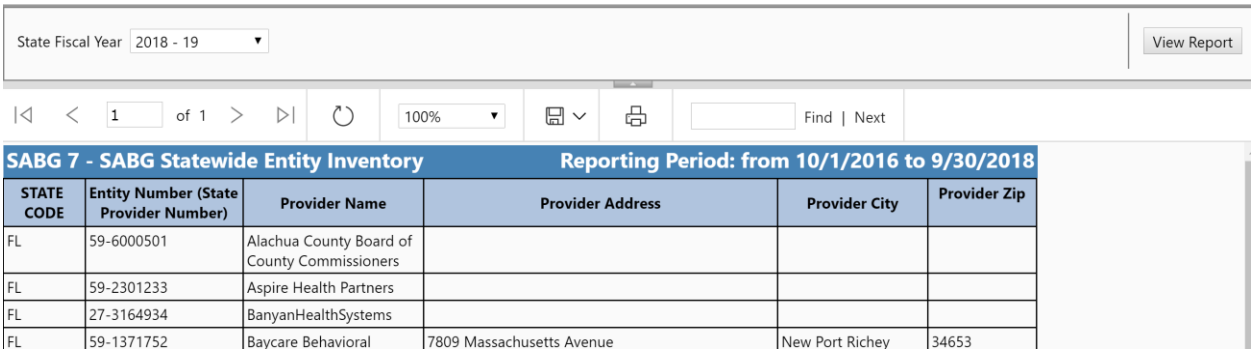
- I-BHS ID

- Area Served (Statewide or Sub-State Planning Area)

Source of Funds columns (A – F): All Block Grant Funds, Prevention (other than Primary Prevention) and Treatment Services, Pregnant Women and Women with Dependent Children, Primary Prevention, Early Intervention Services for HIV, Syringe Services Program

Details of the three runs:

- First run: Default values and there is not multi select. Response time (4 seconds)
- Second Run: Previous fiscal year. Response time (3 secs)
- Third Run: Previous fiscal year. Response time (3 secs)



SABG 7 - SABG Statewide Entity Inventory **Reporting Period: from 10/1/2016 to 9/30/2018**

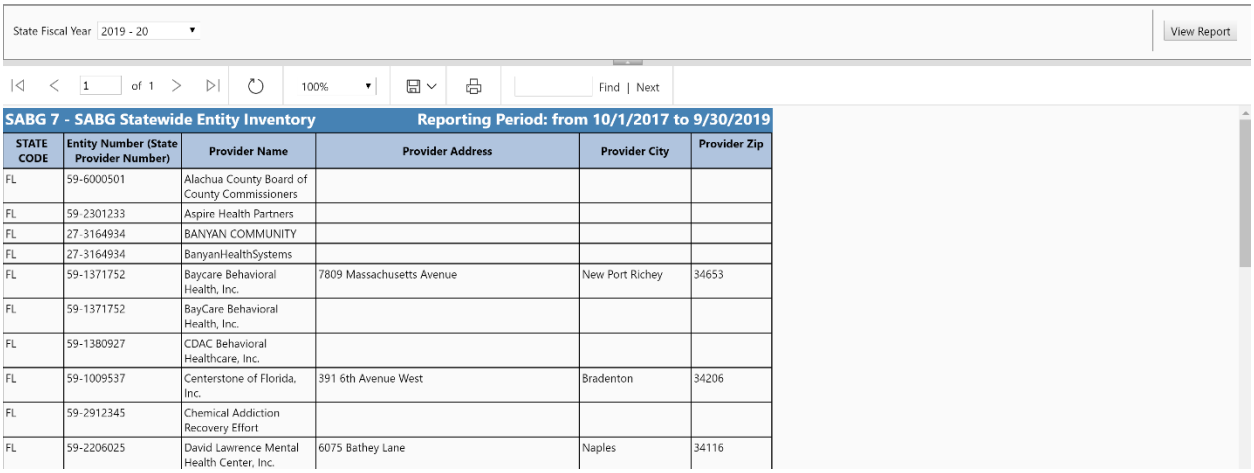
STATE CODE	Entity Number (State Provider Number)	Provider Name	Provider Address	Provider City	Provider Zip
FL	59-6000501	Alachua County Board of County Commissioners			
FL	59-2301233	Aspire Health Partners			
FL	27-3164934	BanyanHealthSystems			
FL	59-1371752	Baycare Behavioral	7809 Massachusetts Avenue	New Port Richey	34653

FIGURE 56 - SABG7 - SABG STATEWIDE ENTITY INVENTORY REPORT EXECUTION SAMPLE

Observations:

- Only a single year can be selected.
  - **Recommendation:** Select multiple years capabilities.
- Headings always need to be visible when scrolling down through the report details

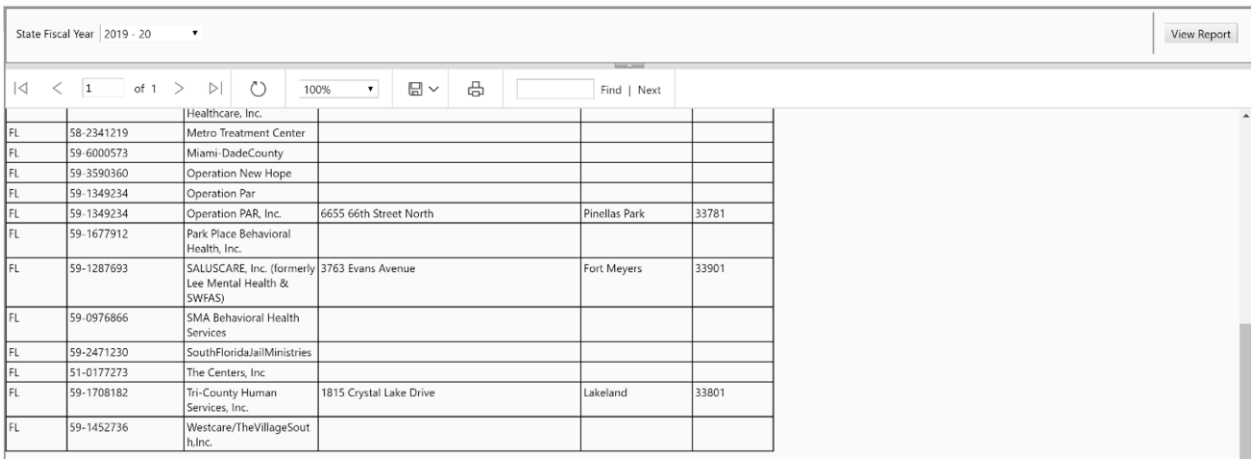
Sample report:



SABG 7 - SABG Statewide Entity Inventory **Reporting Period: from 10/1/2017 to 9/30/2019**

STATE CODE	Entity Number (State Provider Number)	Provider Name	Provider Address	Provider City	Provider Zip
FL	59-6000501	Alachua County Board of County Commissioners			
FL	59-2301233	Aspire Health Partners			
FL	27-3164934	BANYAN COMMUNITY			
FL	27-3164934	BanyanHealthSystems			
FL	59-1371752	Baycare Behavioral Health, Inc.	7809 Massachusetts Avenue	New Port Richey	34653
FL	59-1371752	BayCare Behavioral Health, Inc.			
FL	59-1380927	CDAC Behavioral Healthcare, Inc.			
FL	59-1009537	Centerstone of Florida, Inc.	391 6th Avenue West	Bradenton	34206
FL	59-2912345	Chemical Addiction Recovery Effort			
FL	59-2206025	David Lawrence Mental Health Center, Inc.	6075 Bathey Lane	Naples	34116

FIGURE 57 - SABG7 - SABG STATEWIDE ENTITY INVENTORY REPORT OBSERVATIONS (1 OF 2)



		Healthcare, Inc.			
FL	58-2341219	Metro Treatment Center			
FL	59-6000573	Miami-DadeCounty			
FL	59-3590360	Operation New Hope			
FL	59-1349234	Operation Par			
FL	59-1349234	Operation PAR, Inc.	6655 66th Street North	Pinellas Park	33781
FL	59-1677912	Park Place Behavioral Health, Inc.			
FL	59-1287693	SALUSCARE, Inc. (formerly Lee Mental Health & SWFAS)	3763 Evans Avenue	Fort Meyers	33901
FL	59-0976866	SMA Behavioral Health Services			
FL	59-2471230	SouthFloridaJailMinistries			
FL	51-0177273	The Centers, Inc			
FL	59-1708182	Tri-County Human Services, Inc.	1815 Crystal Lake Drive	Lakeland	33801
FL	59-1452736	Westcare/TheVillageSouth,Inc.			

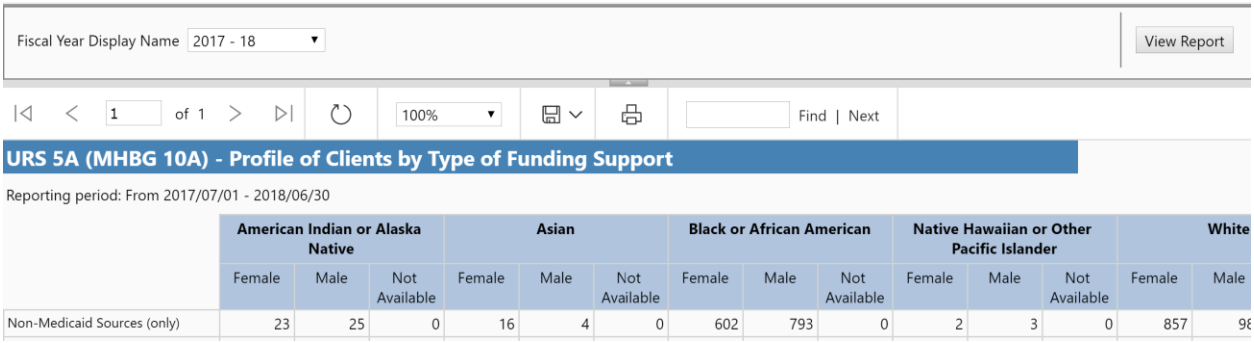
FIGURE 58 - SABG7 - SABG STATEWIDE ENTITY INVENTORY REPORT OBSERVATIONS (2 OF 2)

URS 5A (MHBG10A) - Profile Of Clients By Type Of Funding Support Report

The purpose of this report is to provide the number of mental health clients served by race and Medicaid coverage.

Details of the three runs:

- First run: Default values and there is not multi select. Response time (4 seconds)
- Second Run: Previous fiscal year. Response time (3 secs)
- Third Run: Previous fiscal year. Response time (3 secs)



URS 5A (MHBG 10A) - Profile of Clients by Type of Funding Support

Reporting period: From 2017/07/01 - 2018/06/30

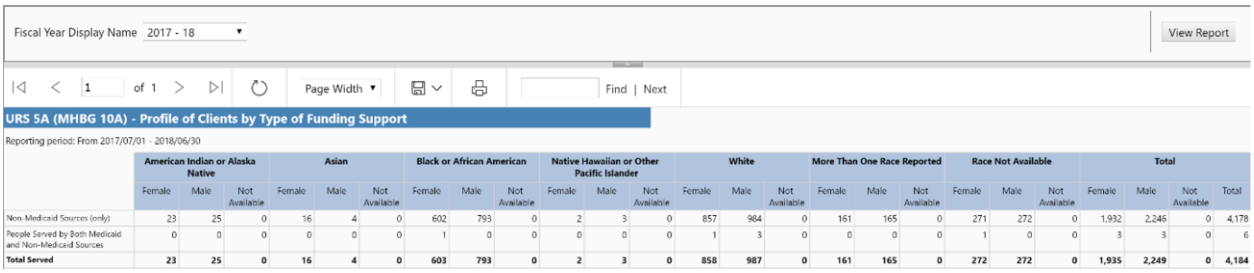
	American Indian or Alaska Native			Asian			Black or African American			Native Hawaiian or Other Pacific Islander			White	
	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Female	Male
Non-Medicaid Sources (only)	23	25	0	16	4	0	602	793	0	2	3	0	857	98

FIGURE 59 - URS 5A (MHBG 10A) - PROFILE OF CLIENTS BY TYPE OF FUNDING SUPPORT REPORT EXECUTION SAMPLE

Observations:

- Fiscal year Display Name: (No multiple selection)
- Headings always need to be visible when scrolling down through the report details

Sample report:



	American Indian or Alaska Native			Asian			Black or African American			Native Hawaiian or Other Pacific Islander			White			More Than One Race Reported			Race Not Available			Total			
	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Total
Non-Medicaid Sources (only)	23	25	0	16	4	0	602	793	0	2	3	0	857	984	0	161	165	0	271	272	0	1,932	2,245	0	4,178
People Served by Both Medicaid and Non-Medicaid Sources	0	0	0	0	0	0	1	0	0	0	0	0	1	3	0	0	0	0	1	0	0	3	3	0	6
Total Served	**23**	**25**	**0**	**16**	**4**	**0**	**603**	**793**	**0**	**2**	**3**	**0**	**858**	**987**	**0**	**161**	**165**	**0**	**272**	**272**	**0**	**1,935**	**2,249**	**0**	**4,184**

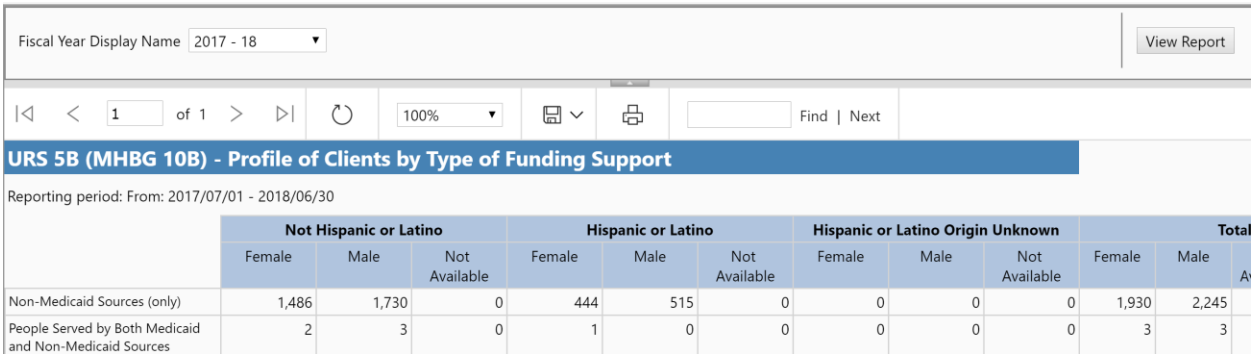
FIGURE 60 - URS 5A (MHBG 10A) - PROFILE OF CLIENTS BY TYPE OF FUNDING SUPPORT REPORT OBSERVATIONS

URS 5B (MHBG10B) - Profile Of Clients By Type Of Funding Support Report

The purpose of this report is to provide the number of mental health clients served by different ethnicities and by Medicaid coverage categories.

Details of the three runs:

- First run: Default values and there is not multi select. Response time (4 seconds)
- Second Run: Previous fiscal year. Response time (3 secs)
- Third Run: Previous fiscal year. Response time (3 secs)



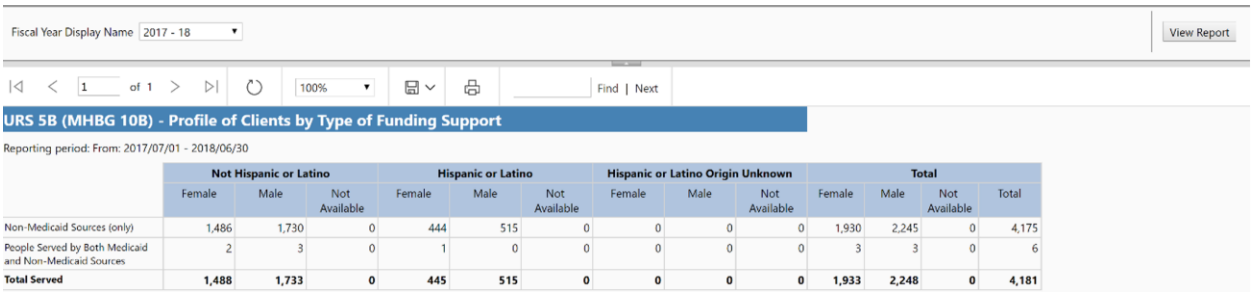
	Not Hispanic or Latino			Hispanic or Latino			Hispanic or Latino Origin Unknown			Total		
	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Female	Male	A
Non-Medicaid Sources (only)	1,486	1,730	0	444	515	0	0	0	0	1,930	2,245	
People Served by Both Medicaid and Non-Medicaid Sources	2	3	0	1	0	0	0	0	0	3	3	

FIGURE 61 - URS 5B (MHBG 10A) - PROFILE OF CLIENTS BY TYPE OF FUNDING SUPPORT REPORT EXECUTION SAMPLE

Observations:

- Fiscal year Display Name: (No multiple selection)
- Headings always need to be visible when scrolling down through the report details

Sample report:



	Not Hispanic or Latino			Hispanic or Latino			Hispanic or Latino Origin Unknown			Total			
	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Female	Male	Not Available	Total
Non-Medicaid Sources (only)	1,486	1,730	0	444	515	0	0	0	0	1,930	2,245	0	4,175
People Served by Both Medicaid and Non-Medicaid Sources	2	3	0	1	0	0	0	0	0	3	3	0	6
Total Served	**1,488**	**1,733**	**0**	**445**	**515**	**0**	**0**	**0**	**0**	**1,933**	**2,248**	**0**	**4,181**

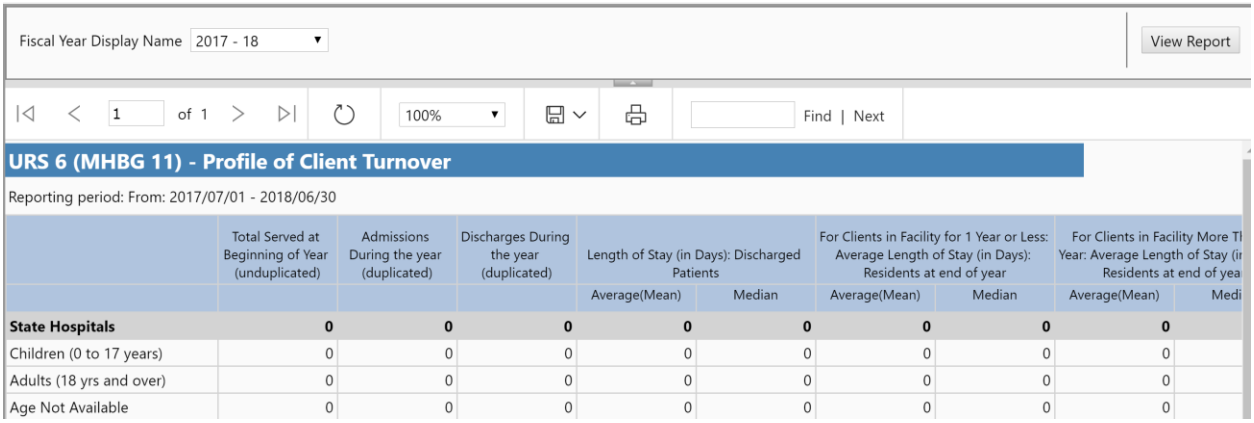
FIGURE 62 - URS 5B (MHBG 10A) - PROFILE OF CLIENTS BY TYPE OF FUNDING SUPPORT REPORT OBSERVATIONS

URS 6 (MHBG 11) - Profile of Client Turnover Report

The purpose of this report is to reflect the client flow and turnover by service settings (state hospitals, other psychiatric inpatient, residential treatment centers and community programs) and age groups.

Details of the three runs:

- First run: Default values and there is not multi select. Response time (4 seconds)
- Second Run: Previous fiscal year. Response time (3 secs)
- Third Run: Previous fiscal year. Response time (3 secs)



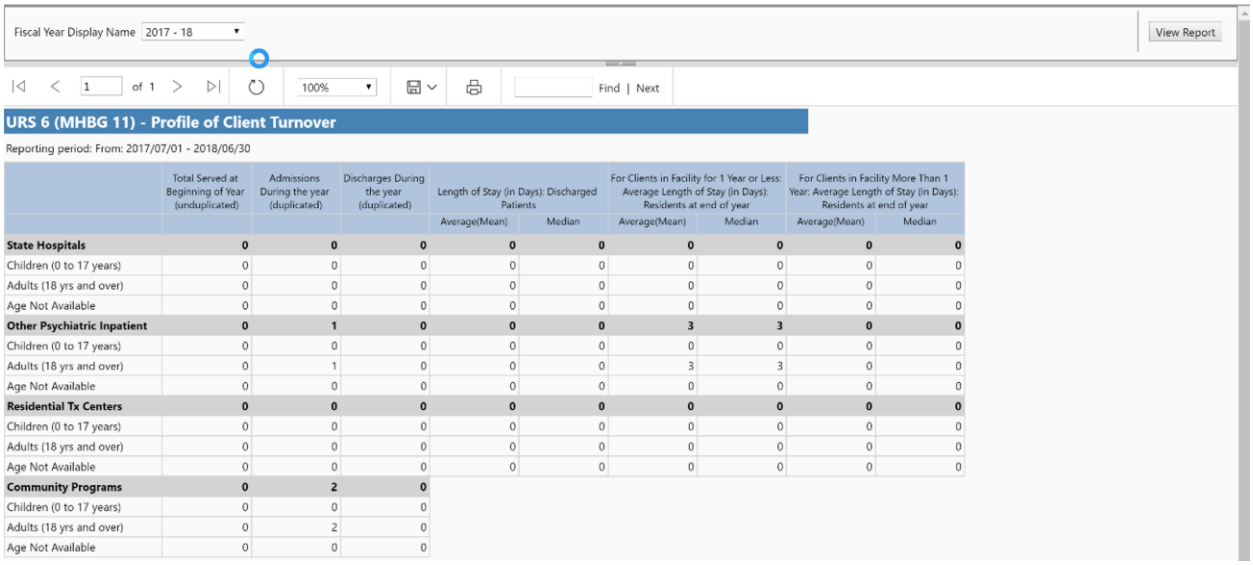
	Total Served at Beginning of Year (unduplicated)	Admissions During the year (duplicated)	Discharges During the year (duplicated)	Length of Stay (in Days): Discharged Patients		For Clients in Facility for 1 Year or Less: Average Length of Stay (in Days): Residents at end of year		For Clients in Facility More Th Year: Average Length of Stay (i Residents at end of yea	
				Average(Mean)	Median	Average(Mean)	Median	Average(Mean)	Medi
State Hospitals	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	
Children (0 to 17 years)	0	0	0	0	0	0	0	0	
Adults (18 yrs and over)	0	0	0	0	0	0	0	0	
Age Not Available	0	0	0	0	0	0	0	0	

FIGURE 63 - URS (MHBG 11) - PROFILE OF CLIENT TURNOVER REPORT EXECUTION SAMPLE

Observations:

- Fiscal year Display Name: (No multiple selection)
- Headings always need to be visible when scrolling down through the report details

Sample report:



	Total Served at Beginning of Year (unduplicated)	Admissions During the year (duplicated)	Discharges During the year (duplicated)	Length of Stay (in Days): Discharged Patients		For Clients in Facility for 1 Year or Less: Average Length of Stay (in Days): Residents at end of year		For Clients in Facility More Than 1 Year: Average Length of Stay (in Days): Residents at end of year	
				Average(Mean)	Median	Average(Mean)	Median	Average(Mean)	Median
State Hospitals	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Children (0 to 17 years)	0	0	0	0	0	0	0	0	0
Adults (18 yrs and over)	0	0	0	0	0	0	0	0	0
Age Not Available	0	0	0	0	0	0	0	0	0
Other Psychiatric Inpatient	**0**	**1**	**0**	**0**	**0**	**3**	**3**	**0**	**0**
Children (0 to 17 years)	0	0	0	0	0	0	0	0	0
Adults (18 yrs and over)	0	1	0	0	0	3	3	0	0
Age Not Available	0	0	0	0	0	0	0	0	0
Residential Tx Centers	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
Children (0 to 17 years)	0	0	0	0	0	0	0	0	0
Adults (18 yrs and over)	0	0	0	0	0	0	0	0	0
Age Not Available	0	0	0	0	0	0	0	0	0
Community Programs	**0**	**2**	**0**						
Children (0 to 17 years)	0	0	0						
Adults (18 yrs and over)	0	2	0						
Age Not Available	0	0	0						

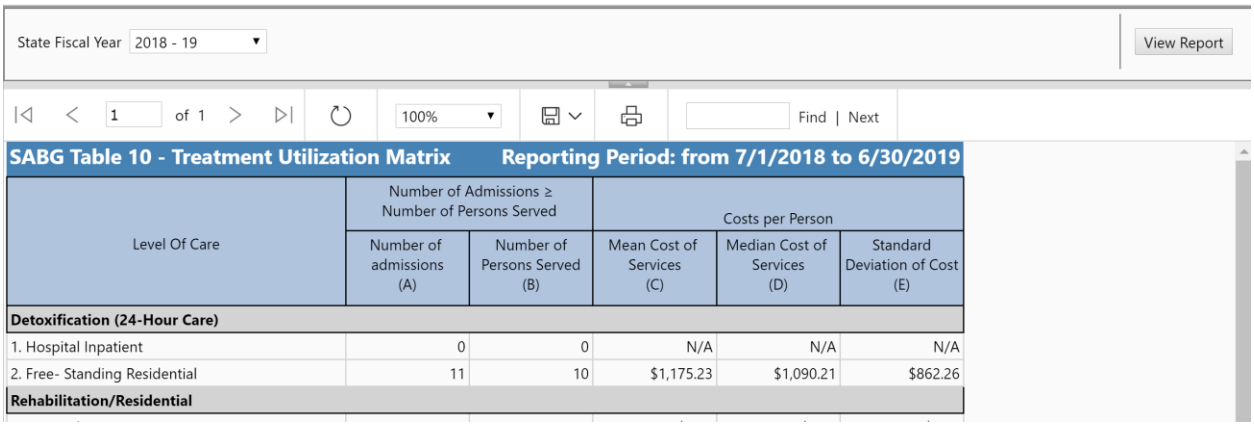
FIGURE 64 - URS (MHBG 11) - PROFILE OF CLIENT TURNOVER REPORT OBSERVATIONS

SABG Table 10 - Treatment Utilization Matrix Report

This report captures the count of persons with initial admissions and subsequent transfer admission(s) to an episode of care in the state fiscal year.

Details of the three runs:

- First run: Default selection does not bring results. After selecting current fiscal year, the response time (3 secs)
- Second run: Previous fiscal year. Response time (3 secs)
- Third run: Previous fiscal year. Response time (3 secs)



SABG Table 10 - Treatment Utilization Matrix **Reporting Period: from 7/1/2018 to 6/30/2019**

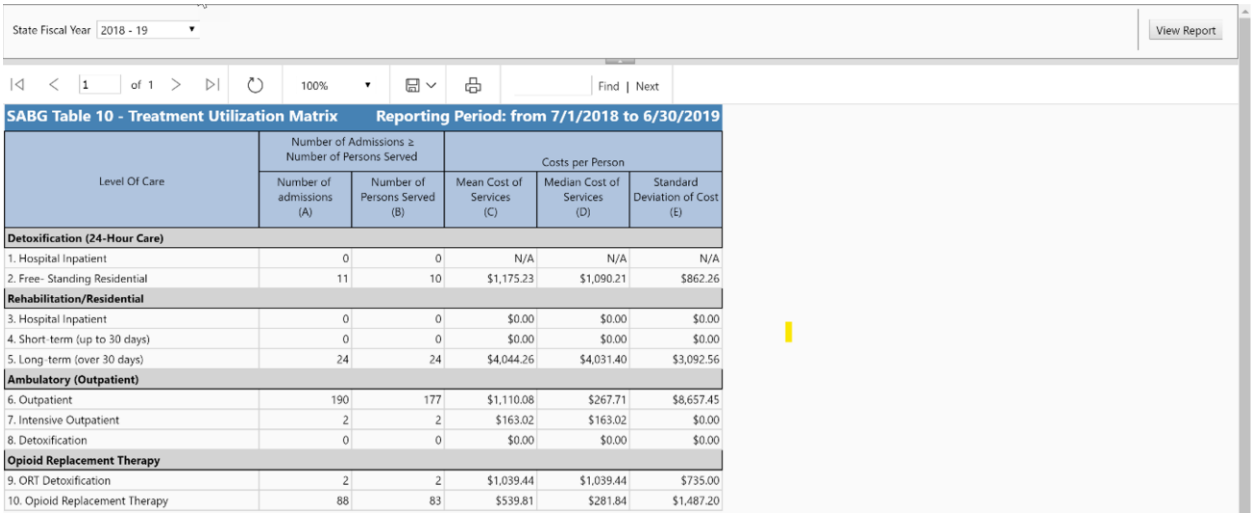
Level Of Care	Number of Admissions ≥ Number of Persons Served		Costs per Person		
	Number of admissions (A)	Number of Persons Served (B)	Mean Cost of Services (C)	Median Cost of Services (D)	Standard Deviation of Cost (E)
Detoxification (24-Hour Care)					
1. Hospital Inpatient	0	0	N/A	N/A	N/A
2. Free- Standing Residential	11	10	$1,175.23	$1,090.21	$862.26
Rehabilitation/Residential					

FIGURE 65 - SABG TABLE 10 - TREATMENT UTILIZATION MATRIX REPORT EXECUTION SAMPLE

Observations:

- Fiscal year Display Name: (No multiple selection)
- Headings always need to be visible when scrolling down through the report details

Sample report:



SABG Table 10 - Treatment Utilization Matrix **Reporting Period: from 7/1/2018 to 6/30/2019**

Level Of Care	Number of Admissions ≥ Number of Persons Served		Costs per Person		
	Number of admissions (A)	Number of Persons Served (B)	Mean Cost of Services (C)	Median Cost of Services (D)	Standard Deviation of Cost (E)
Detoxification (24-Hour Care)					
1. Hospital Inpatient	0	0	N/A	N/A	N/A
2. Free- Standing Residential	11	10	$1,175.23	$1,090.21	$862.26
Rehabilitation/Residential					
3. Hospital Inpatient	0	0	$0.00	$0.00	$0.00
4. Short-term (up to 30 days)	0	0	$0.00	$0.00	$0.00
5. Long-term (over 30 days)	24	24	$4,044.26	$4,031.40	$3,092.56
Ambulatory (Outpatient)					
6. Outpatient	190	177	$1,110.08	$267.71	$8,657.45
7. Intensive Outpatient	2	2	$163.02	$163.02	$0.00
8. Detoxification	0	0	$0.00	$0.00	$0.00
Opioid Replacement Therapy					
9. ORT Detoxification	2	2	$1,039.44	$1,039.44	$735.00
10. Opioid Replacement Therapy	88	83	$539.81	$281.84	$1,487.20

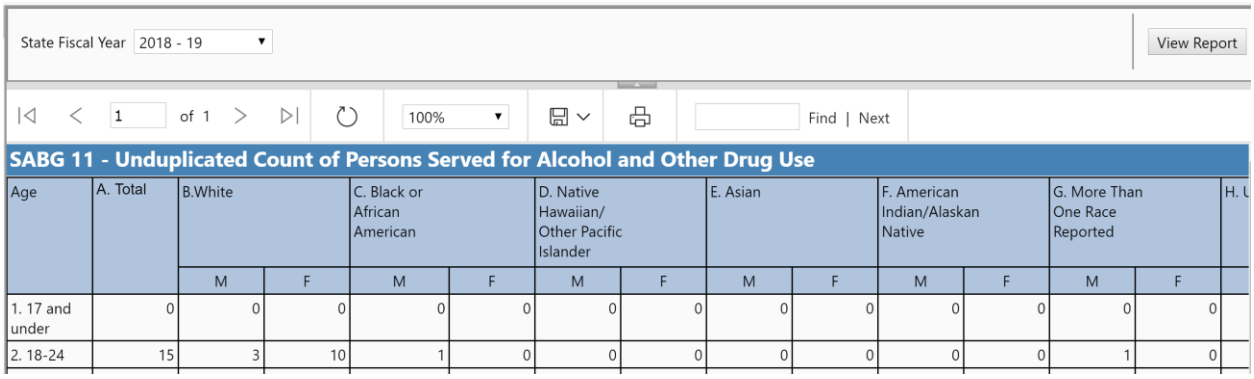
FIGURE 66 - SABG TABLE 10 - TREATMENT UTILIZATION MATRIX REPORT EXECUTION SAMPLE

SABG 11 - Unduplicated Count of Persons Served for Alcohol and Other Drug Use Report

This report provides an aggregate profile of the unduplicated number of clients served through the SABG fund for the state fiscal year.

Details of the three runs:

- First run: Default selection does not bring results. After selecting current fiscal year, the response time (4 secs)
- Second run: Previous fiscal year. Response time (2 secs)
- Third run: Previous fiscal year. Response time (2 secs)



SABG 11 - Unduplicated Count of Persons Served for Alcohol and Other Drug Use

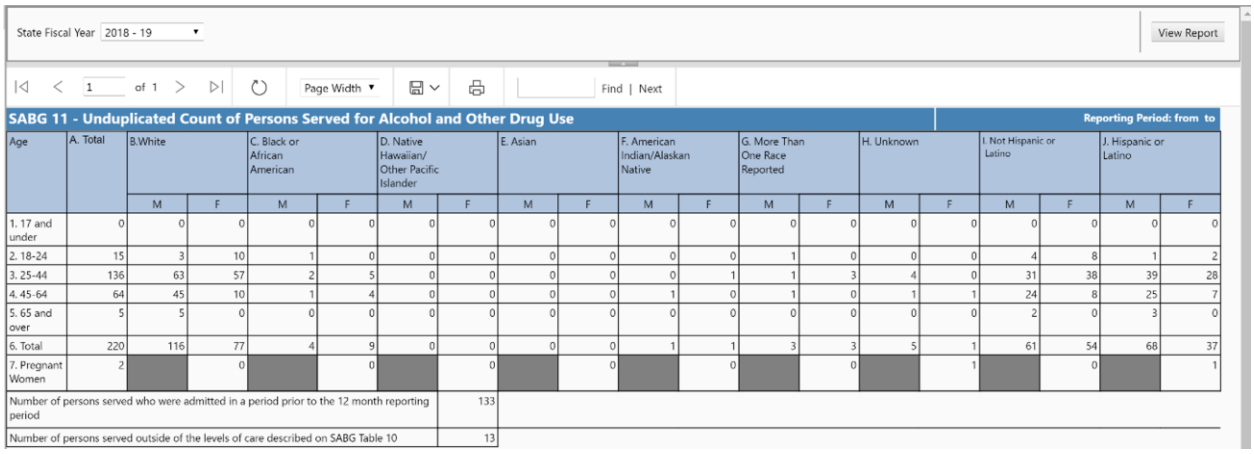
Age	A. Total	B.White		C. Black or African American		D. Native Hawaiian/ Other Pacific Islander		E. Asian		F. American Indian/Alaskan Native		G. More Than One Race Reported		H. U
		M	F	M	F	M	F	M	F	M	F	M	F	
1. 17 and under	0	0	0	0	0	0	0	0	0	0	0	0	0	
2. 18-24	15	3	10	1	0	0	0	0	0	0	0	1	0	

FIGURE 67 - SABG 11 - UNDUPLICATED COUNT OF PERSONS SERVED FOR ALCOHOL AND OTHER DRUG USE REPORT SAMPLE

Observations:

- Fiscal year Display Name: (No multiple selection)
- Headings always need to be visible when scrolling down through the report details

Sample report:



SABG 11 - Unduplicated Count of Persons Served for Alcohol and Other Drug Use Reporting Period: from to

Age	A. Total	B.White		C. Black or African American		D. Native Hawaiian/ Other Pacific Islander		E. Asian		F. American Indian/Alaskan Native		G. More Than One Race Reported		H. Unknown		I. Not Hispanic or Latino		J. Hispanic or Latino	
		M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F
1. 17 and under	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2. 18-24	15	3	10	1	0	0	0	0	0	0	0	1	0	0	0	4	8	1	2
3. 25-44	136	63	57	2	5	0	0	0	0	0	1	1	3	4	0	31	38	39	28
4. 45-64	64	45	10	1	4	0	0	0	0	1	0	1	0	1	1	24	8	25	7
5. 65 and over	5	5	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	3	0
6. Total	220	116	77	4	9	0	0	0	0	1	1	3	3	5	1	61	54	68	37
7. Pregnant Women	2		0		0		0		0		0		0		1		0		1
Number of persons served who were admitted in a period prior to the 12 month reporting period							133												
Number of persons served outside of the levels of care described on SABG Table 10							13												

FIGURE 68 - SABG 11 - UNDUPLICATED COUNT OF PERSONS SERVED FOR ALCOHOL AND OTHER DRUG USE SAMPLE

URS 12 (MHBG 12) - State Mental Health Agency Profile Report

The purpose of this report is to provide calculated answers to question 2 and 3 of URS Table 12 (MHBG Table 12). Note that question 1, 4, 5 and 6 of the URS/MHBG table will not be included because FASAMS does not have the required information.

Details of the three runs:

- First run: Default selection. Response time (4 secs)
- Second run: Previous fiscal year. Response time (2 secs)
- Third run: Previous fiscal year. Response time (2 secs)

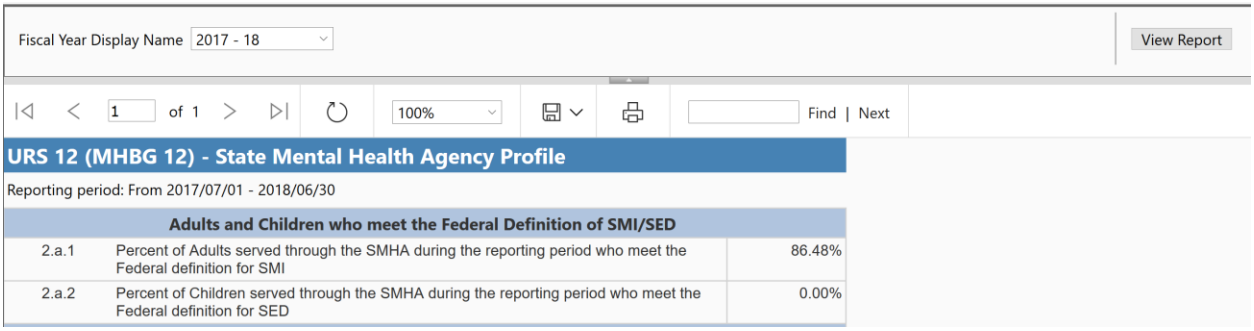


FIGURE 69 - URS 12 (MHBG 12) - STATE MENTAL HEALTH AGENCY PROFILE REPORT EXECUTION SAMPLE

Observations:

- "Fiscal year display name" values are ordered descending (Most of the reports have it ascending)

Sample report:

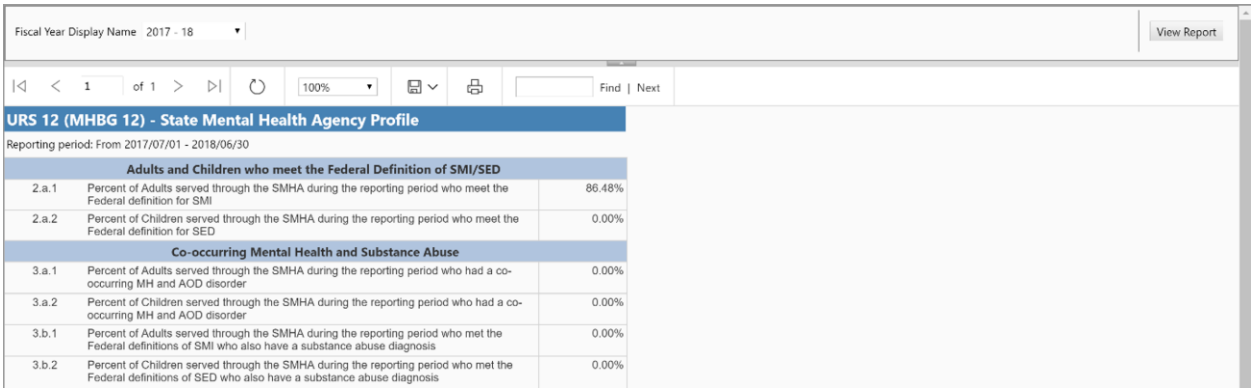


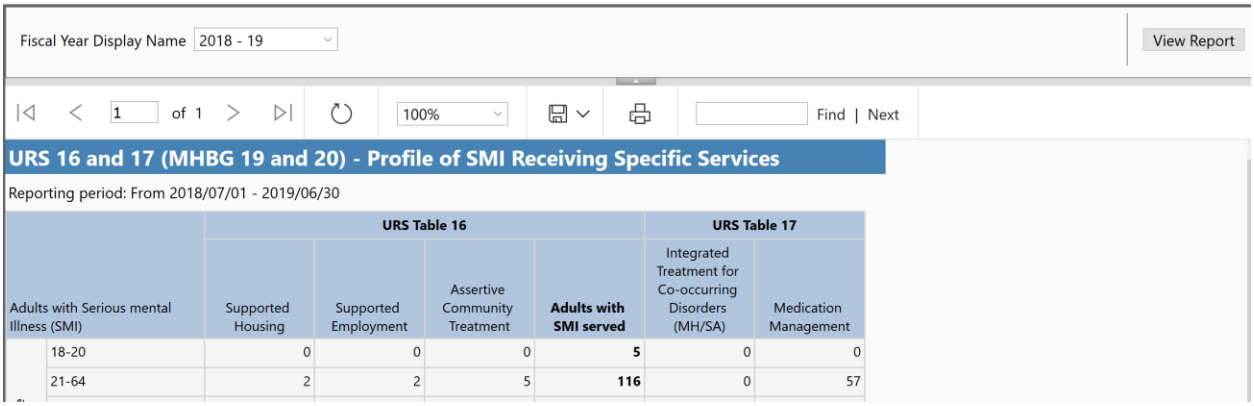
FIGURE 70 - URS 12 (MHBG 12) - STATE MENTAL HEALTH AGENCY PROFILE REPORT OBSERVATIONS

URS 16 and 17 (MHBG 19 and 20) - Profile of SMI Receiving Specific Services Report

The purpose of this report is to provide a profile of serious mental illness (SMI) adults receiving specific evidence-based practices (EBP) in the reporting fiscal year. This report will combine URS Table 16 and 17. Serious Emotional Disturbances (SED) children columns from Table 16 will not be included, because FASAMS does not store children's EBP program data, which is required for reporting.

Details of the three runs:

- First run: Default selection. Response time (4 secs)
- Second run: Previous fiscal year. Response time (2 secs)
- Third run: Previous fiscal year. Response time (2 secs)



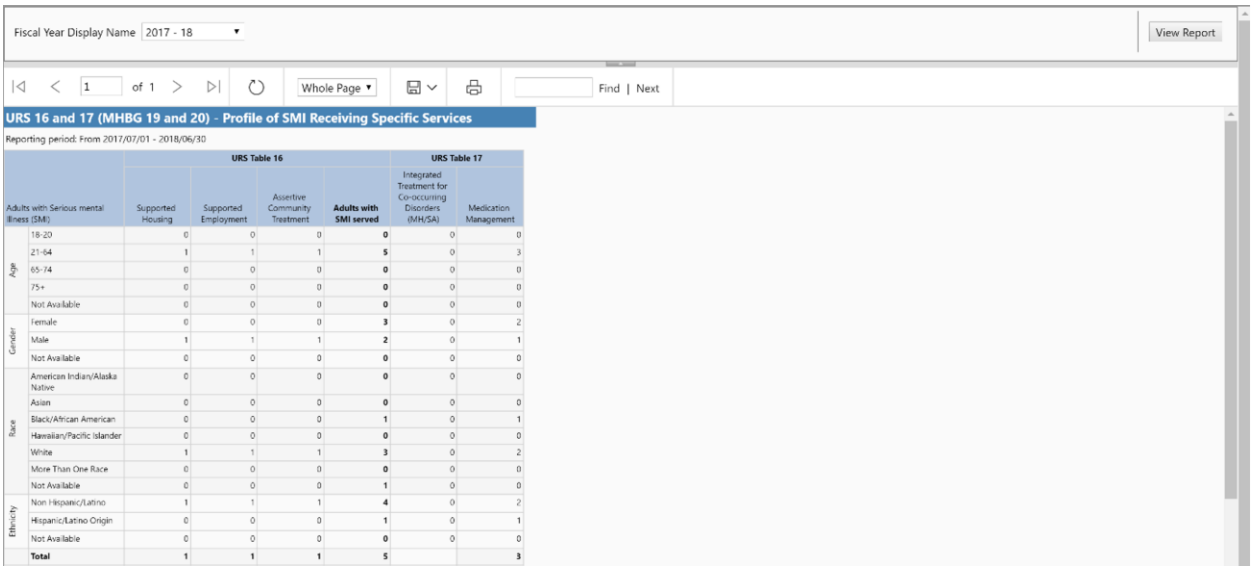
Adults with Serious mental Illness (SMI)	URS Table 16				URS Table 17	
	Supported Housing	Supported Employment	Assertive Community Treatment	**Adults with SMI served**	Integrated Treatment for Co-occurring Disorders (MH/SA)	Medication Management
18-20	0	0	0	**5**	0	0
21-64	2	2	5	**116**	0	57

FIGURE 71 - URS 16 AND 17 (MHBG 19 AND 20) - PROFILE OF SMI RECEIVING SERVICE REPORT EXECUTION SAMPLE

Observations:

- "Fiscal year display name" values are ordered descending (Must of the reports have it ascending) - (No multiple selection)
- Headings always need to be visible when scrolling down through the report details

Sample report:



	Adults with Serious mental Illness (SMI)	URS Table 16				URS Table 17	
		Supported Housing	Supported Employment	Assertive Community Treatment	**Adults with SMI served**	Integrated Treatment for Co-occurring Disorders (MH/SA)	Medication Management
Age	18-20	0	0	0	**0**	0	0
	21-64	1	1	1	**5**	0	3
	65-74	0	0	0	**0**	0	0
	75+	0	0	0	**0**	0	0
	Not Available	0	0	0	**0**	0	0
Gender	Female	0	0	0	**3**	0	2
	Male	1	1	1	**2**	0	1
	Not Available	0	0	0	**0**	0	0
Race	American Indian/Alaska Native	0	0	0	**0**	0	0
	Asian	0	0	0	**0**	0	0
	Black/African American	0	0	0	**1**	0	1
	Hawaiian/Pacific Islander	0	0	0	**0**	0	0
	White	1	1	1	**3**	0	2
	More Than One Race	0	0	0	**0**	0	0
	Not Available	0	0	0	**1**	0	0
Ethnicity	Non Hispanic/Latino	1	1	1	**4**	0	2
	Hispanic/Latino Origin	0	0	0	**1**	0	1
	Not Available	0	0	0	**0**	0	0
	Total	**1**	**1**	**1**	**5**		**3**

FIGURE 72 - URS 16 AND 17 (MHBG 19 AND 20) – PROFILE OF SMI RECEIVING SERVICE REPORT OBSERVATIONS

Financial Reports

Provider Expenditure Validation (For DCF) Report

[No descriptive information was found by the team in the documentation to provide a report description]

Details of the three runs are:

- First run: No default values selected. After selecting random values. Response time (6 secs)
- Second run: Selecting random values. Response time (3 secs)
- Third run: Selecting random values. Response time (8 secs)

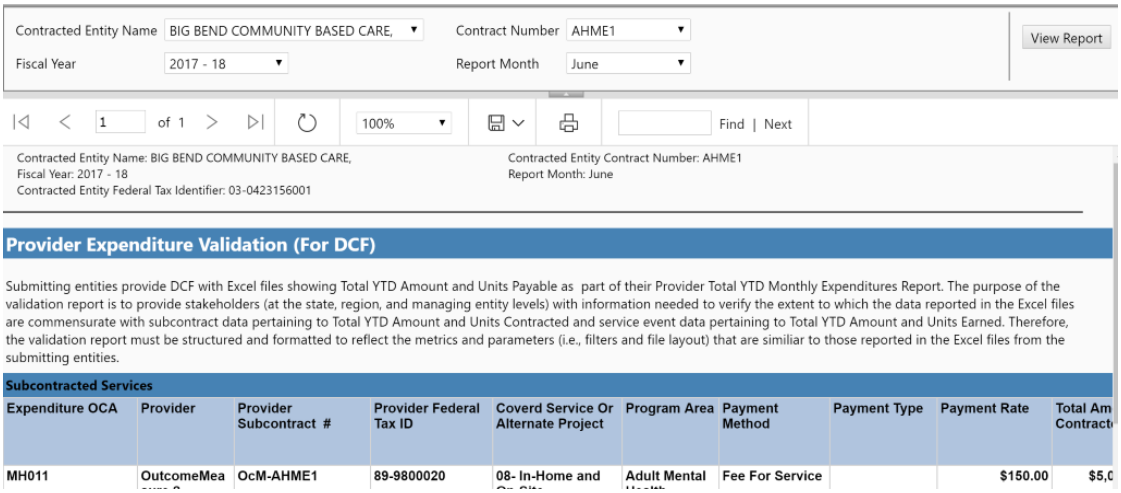


FIGURE 73 - PROVIDER EXPENDITURE VALIDATION (FOR DCF) REPORT EXECUTION SAMPLE

Observations:

- No multiple selection available in any field.
- Changing the selection from first option, clears out other fields which forces user to select everything again.

Provider Expenditure Validation (For Quality Assurance Purposes) Report

[No descriptive information was found by the team in the documentation to provide a report description]

Details of the three runs are:

- First run: No default values selected. After selecting random values. Response time (6 secs)
- Second run: Selecting random values. Response time (3 secs)
- Third run: Selecting random values. Response time (8 secs)

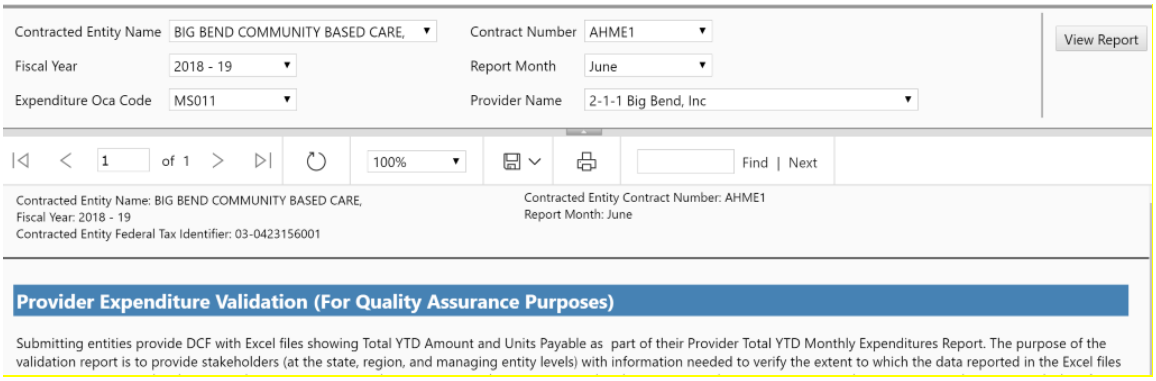


FIGURE 74 - PROVIDER EXPENDITURE VALIDATION (FOR DCF) REPORT OBSERVATIONS

Observations:

- No multiple selection available in any field.
- Changing the selection from first option, clears out other fields which forces user to select everything again.

Job Submission Performance Reports

Failed Records Report

The purpose of the report is to provide submitting entities and DCF users and administrators the ability to understand the failed records and corresponding error messages for each submitting entity, across jobs.

Details of the three runs:

- First run: Date range selected is 4/1/19-4/30/19. All Data Set types selected, and default of Error fix timeframe of 60 days selected. Response time (28 secs)
- Second run: Date range selected is 7/1/18 - 6/30/19. All Data Set types selected, and default of Error fix timeframe of 60 days selected. Response time (after 8 minutes, the system displayed a blank screen and never went back to the report – see the blank screen below)

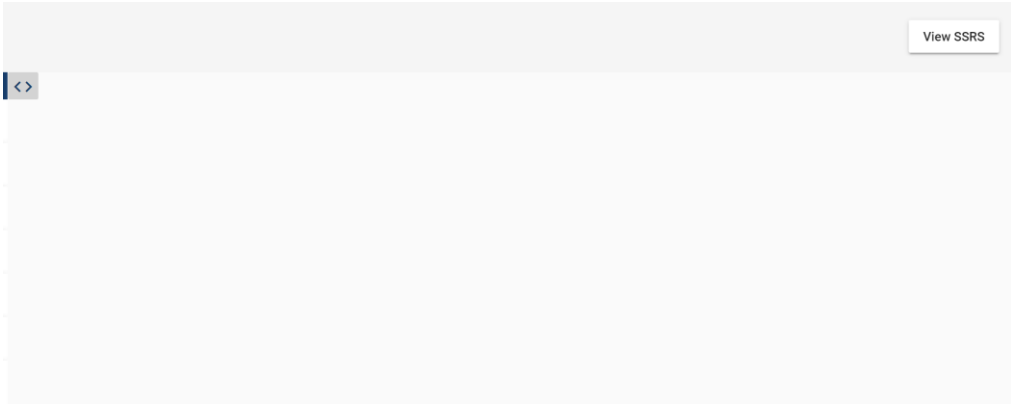


FIGURE 75 - FAILED RECORDS REPORT BLANK SCREEN RESULT

- Ran the second run in SSRS and it finished in 17:32.
- Third run: Same criteria as first run. Response time (28 secs)

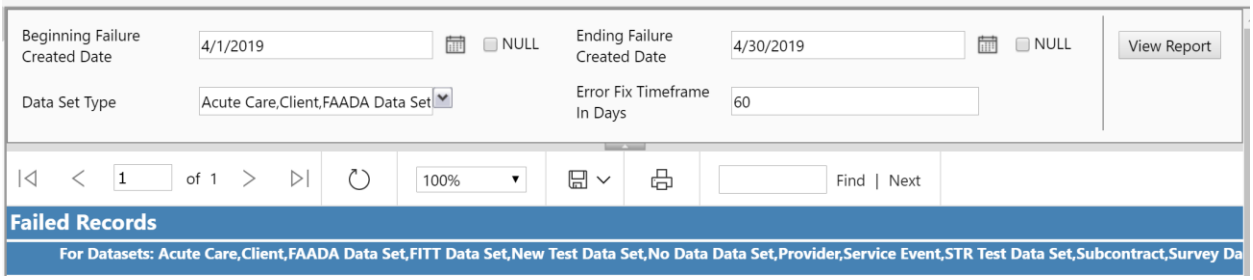


FIGURE 76 - FAILED RECORDS REPORT EXECUTION SAMPLE

Overall Job Performance Report

The purpose of the report is to provide submitting entities and DCF users and administrators the ability to understand the overall trend of their job performance, which would be designed to show whether or not submitting entities are improving with their job submissions over time.

Details of the three runs:

- First run: Runs automatically for default options. Date range is 1/1/2019 – 12/31/2019, All Users, All datasets. Response time (1 secs)
- Second run: All options selected for multiselect fields. Response time (5 secs)

- Third run: All options with Date range of 1/1/1900-12/31/2019. Response time (after 12:33 minutes, the system displayed a blank screen and never went back to the report)

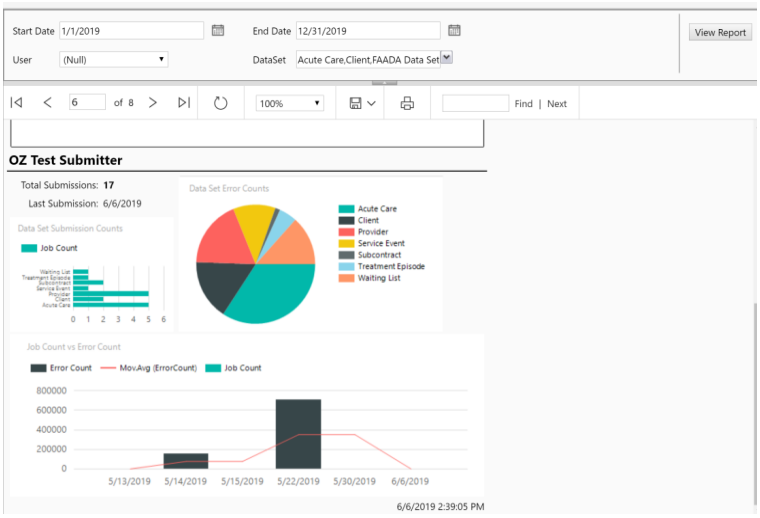


FIGURE 77 - OVERALL JOB PERFORMANCE REPORT EXECUTION SAMPLE

Observations:

- Date range should be limited to realistic dates. For example, there will be no records dating back to 1900 and we cannot have submissions in the future.
    - Recommendation: Modify the drop-down list to only include valid dates.

Submission Summary Report

[No descriptive information was found by the team in the documentation to provide a report description]

Details of the three runs:

- First run: No default options. Date range is 1/1/2019 – 12/31/2019, All Providers selected. Response time (35 secs)
- Second run: Same selection from first run. Response time (25 secs)
- Third run: Same selection from first run. Response time (25 secs

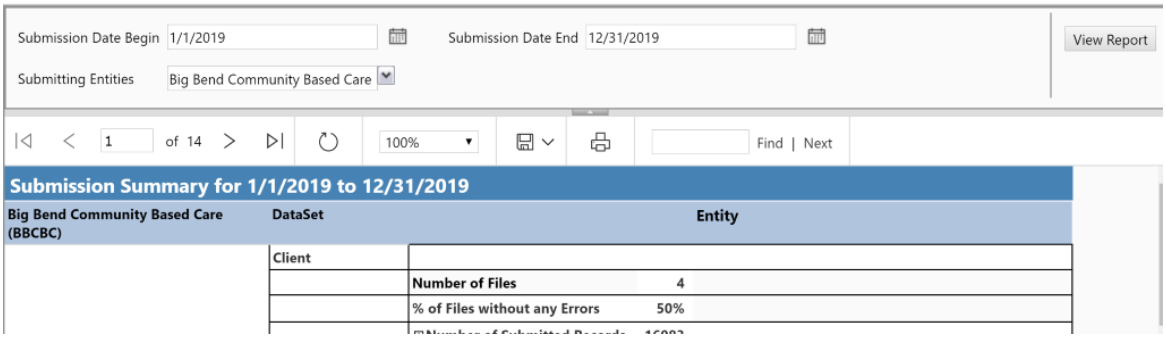


FIGURE 78 - SUBMISSION SUMMARY REPORT EXECUTION SAMPLE

Observations:

- Multi select is available to all fields

- Date range should be limited to realistic dates. For example, there will be no records dating back to 1900 and we cannot have submissions in the future.
    - Recommendation: Modify the drop-down list to only include valid dates.

Submission Rollup Report

[No descriptive information was found by the team in the documentation to provide a report description]

Details of the three runs:

- First run: No default options. Date range is 7/1/2018 – 6/30/2019, All Providers selected. Response time (8:30 minutes)
- Second run: Same selection from first run. Response time (8:06 minutes)
- Third run: Same selection from first run. Response time (8:30 minutes)

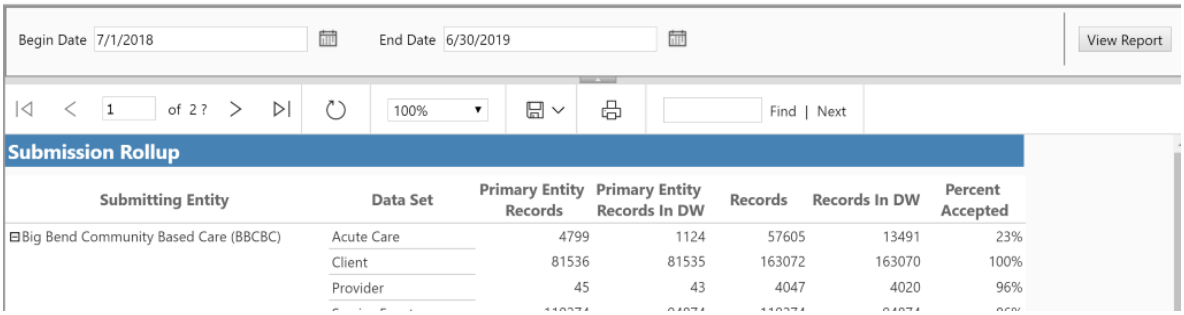


FIGURE 79 - SUBMISSION ROLLUP REPORT EXECUTION SAMPLE

Observations:

- There isn't a function to isolate a specific ME
    - Recommendation: Add a function to filter by ME
- Date range should be limited to realistic dates. For example, there will be no records dating back to 1900 and we cannot have submissions in the future.
    - Recommendation: Modify the drop-down list to only include valid dates.

9.4. Administration Module

The administration module is where submitting entities, users, messages, groups and roles are created, modified and deleted. The Administration page allows an end user with administrative rights to:

- Submitting Entities: View, create, delete and search submitting entities
- Messages: View and create system messages
- Users: View, create, reset password, lock individual accounts, search users and control a global lock out feature
- Groups: View, create and search groups
- Roles: View, create, delete and search roles.

9.4.1. Challenges with the administration module

- **Username and email cannot be changed.** We saw that a username cannot be edited or changed in the system. If a user has a name or email change, a new account must be created, and history is lost.
    - Recommendation: Add a function to rename a user account and change email address or to link/migrate data to the new user account.

9.4.2. General User Experience Recommendations for the Administration Module

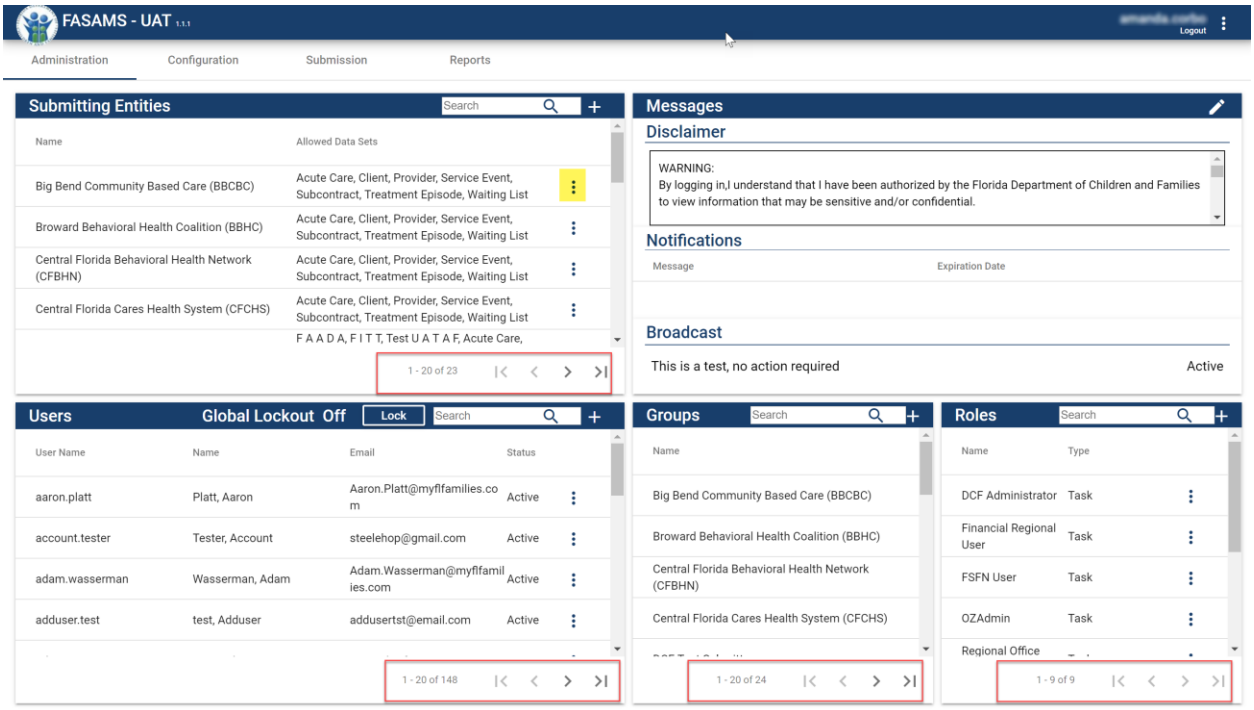


FIGURE 80 - ADMINISTRATION LANDING PAGE

- **Observation**: This page looks very busy and can be difficult to read
  - Recommendation: Create sub-navigation under the 'Administration' section of the website for each of the areas (Submitting Entities, Messages, Users, Groups and Roles) that will take the user to a page to display each of the areas in the full viewing area of the page.
- **Observation**: Additional functionality for each area is hidden behind ellipse menus so the end user does not know what functionality is available to them until they click each to get a menu of options
  - Recommendation: Employing the above recommendation would provide enough space to leave additional options to be listed directly on canvas.
- **Observation**: Most sections require scrolling as well as pagination to view the lists within each section.
  - Recommendation: Using the full page will increase viewing area and illuminate the need for scroll bars as well as provide opportunities to add more functionality on canvas to expose all options available to the end user.

9.5. Configuration Module

The configuration module is where the vocabulary versions, rules and dynamic data sets are created and modified. The vocabulary function allows the MEs to map the values in FASAMS to the values that their source systems already use so that they do not have to add programming to change the values.

During none of the discussions or interviews were issues with the configuration module raised.

9.5.1. General User Experience Recommendations for the Configuration Module

The Configuration page allows an end user with administrative rights to:

- Vocabulary Versions: View, Create and Search Vocabulary Versions
- Dynamic Data Sets: View, Create and Search dynamic data sets
- Rules: View, Create and Search rules and parameters

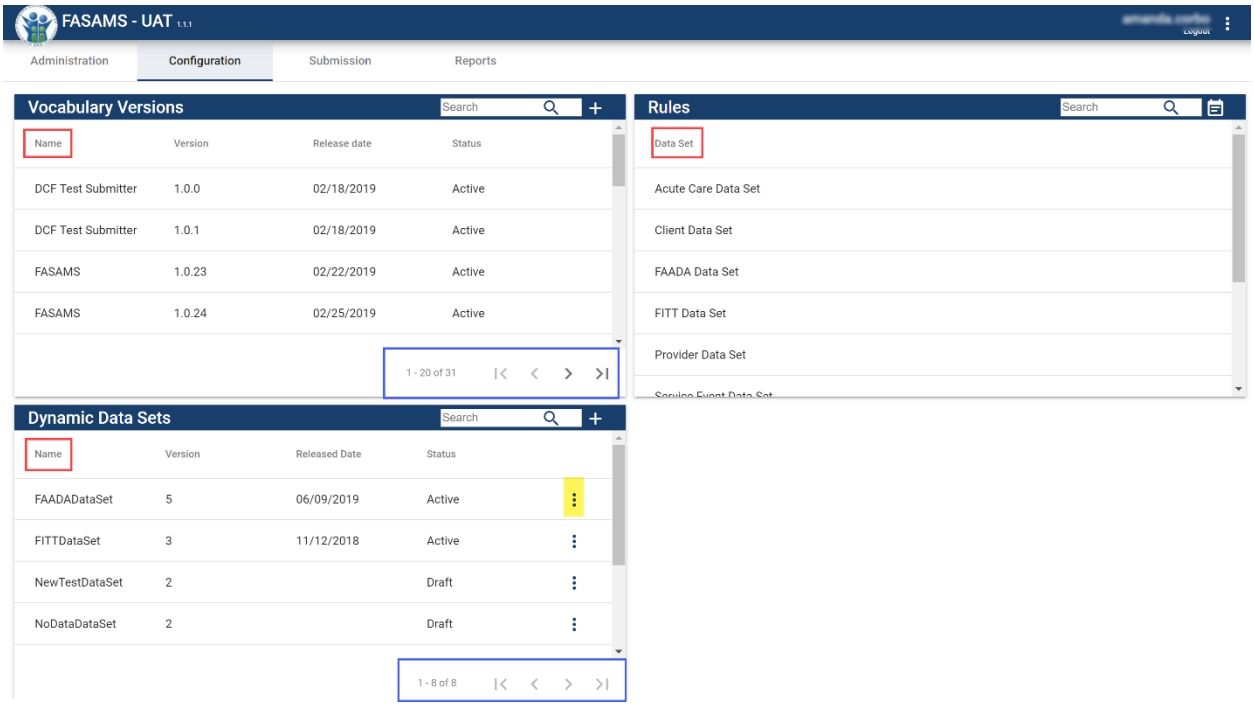


FIGURE 81 - CONFIGURATION LANDING PAGE

- **Observations:** Too many sections as well as content on this page make this page somewhat difficult to read.
  - Recommendation: Add sub menus to the primary navigation for Configuration that will take the end user directly to the respective page for Vocabulary Versions, Rules and Dynamic Data Sets. This will provide a larger viewing area for the content as well as reduce scrolling and pagination in comparison to how it is currently presented.
- **Observations:** Additional functionality for Dynamic Data Sets are hidden behind the ellipse menus in the last column. As a result, the end user does not know what functionality is available to them until they click each to get a menu of options
  - Recommendation: Employing the above recommendation would provide enough space to leave additional options to be listed directly on canvas and eliminate the need for an ellipse.
- **Observations:** It is not obvious that the 'Name' values in each row are clickable and will take the end user to the details page.
  - Recommendation: Add a color change when the end user hovers over the text to make it obvious that it is a link.
- **Observations:** Most sections require scrolling as well as pagination to view the lists within each section.
  - Recommendation: Using the full page will increase viewing area and eliminate the need for scroll bars as well as provide opportunities to add more functionality on canvas to expose all options available to the end user.

9.6. Data

9.6.1. Observations relating to the data

During our interviews, we heard various concerns about the data that is available in FASAMS.

- **Database that is available for reporting is difficult to use.** We heard that while there is a data warehouse, it seems that the reports run against the OLTP database. This is causing multiple issues:
    - The current implementation does not contain elements in the data handling and storage side that would make easier to build new reports.
    - The team is accessing the production database directly to run reports, which affects system performance.
    - DCF's Data and Reporting team have created some workarounds to speed up the way reports are built but the team recognizes that FASAMS should have been better designed knowing that the outcome was to produce reports to different DCF's departments.
    - Currently, ad-hoc reports are built using SQL queries with multiple joins that may affect performance once more users have access to run reports.
        - Recommendation: Create a reporting database
- **Users need to be able to mark records as "deleted".** After the system was put into production the need for delete/soft-delete functionality was identified as there is no way to mark a record as deleted via the application. The team is currently writing SQL statements to delete/soft-delete data as needed.
    - Recommendation: Add a function to the application that allows users with appropriate security to mark a record as deleted. This will improve data integrity both in the data and on the reports. It will also mitigate the risk inherent when staff is writing SQL statements to delete data.
- **Data becomes orphaned when data is deleted via SQL script.** The way the data is structured in a cascading relationship creates data inconsistency when a soft delete is executed. Some data becomes orphaned and this creates issues with performance and report accuracy.
    - Recommendation: Add a function to the application that allows users with appropriate security to make a record as deleted. This function will also include rules that do not allow orphaned records.
    - Recommendation: Add constraints on the database that will not allow orphaned records.
- **Data does not match provider records.** The data produced by FASAMS reports is not reliable and it does not match with the information provided by the Providers.
    - Recommendation: Perform full QA check of the system. This process should include records originated by the provider and follow the records all the way through the system.
- **Data does not match MEs manual records**. The data from FASAMS is compared to the MEs manual records. At this point in time, the manual records are trusted and not FASAMS.
    - Recommendation: Perform full QA check of the system. This process should include records submitted by the ME and follow the records all the way through the system.
- **Data is inconsistent between providers.** The upload data produces reports that affect the resources distribution because the reports do not represent the real accountability of services served. One of the reasons for this could be the use of "Optionally required fields" that some providers will mark the fields as not required and this generates data inconsistency.
    - Recommendation: Create a process that ensures consistency between providers and MEs.

9.6.2. FASAMS Logical Data Schema/Structure

Managing entities must create XML files to load into FASAMS. There are multiple types of records and the records must be submitted in the correct order, or the import will fail, because records are dependent upon each other. For example, the provider must exist in the system before a client can receive services from that provider. The following section describes the record types and dependencies in a diagram and narrative form.

Data Sets - Datasets represent each data file required for collecting and reporting data on persons served only in state-contracted community substance abuse and mental health Provider agencies, but also in state-contracted or state-operated mental health treatment facilities. This data is needed, at the federal, state and local levels, to answer the management question "who received what services from whom at what cost to achieve what outcomes?"

At the federal level, these data are collected as part of the Block Grant requirements for Treatment Episode Data Sets (TEDS), which includes the National Outcome Measures (NOM) data, the Uniform Reporting System (URS) data, and the Basic Client Information (BCI) data. At the state and local levels, these data are used for reporting various metrics, including performance outcome and output measures required by the Legislature as part of the General Appropriation Act (GAA), as well as quality assurance and quality improvement measures required by DCF for program planning and budgeting, contract monitoring, and various other priority of effort (POE) initiatives.

Dataset Dependency

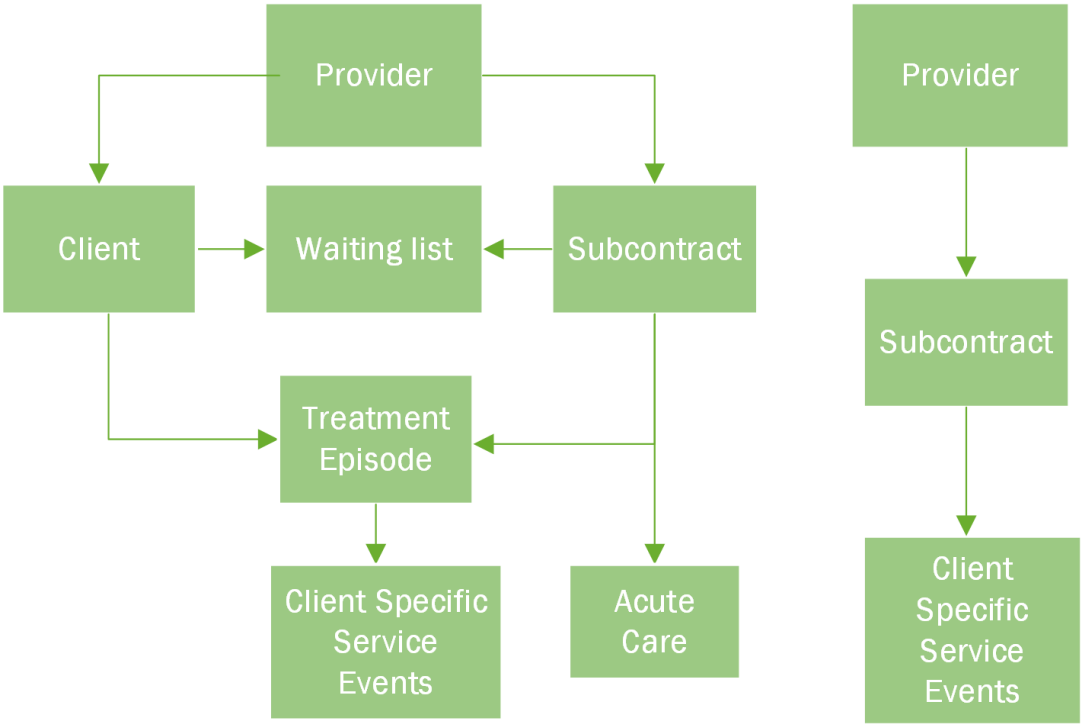


FIGURE 82 - DATASET DEPENDENCY

Provider

Description - Provider represents all agencies that have a contract with or are licensed by DCF for community mental health or substance abuse services, also represents all state treatment facilities that are operated by or have a contract with DCF. Provider data includes organization-level data related to contact persons, as well as identification numbers, names, addresses, and sites of the Provider agencies that are state-contracted or state-operated.

Hierarchy - Provider is the **parent** dataset that must be collected when there is a new Provider site that needs to submit data to FASAMS.

Client

Description - Client represents all individuals receiving substance abuse and/or mental health services whose cost of care is funded, in whole in part, by DCF funds. Client data includes Protected Health Information (PHI) (e.g., names, Social Security Number, date of birth, race, gender, and ethnicity) and other demographic information on each person whose services are funded in part or in whole by DCF.

Hierarchy - Client dataset depends of Provider data and must be collected every time an individual receives services from the Provider and when data that identifies the individual needs to be updated (i.e. address change).

Treatment Episode

Description - Treatment Episode data include individual-level data of each person who meets criteria for enrollment in any mental health or substance abuse priority population group whose services are funded in whole or in part by DCF. It includes various outcomes, evaluations and diagnosis information.

Hierarchy - Treatment Episode depends of Providers and Clients data and must be collected when an individual is admitted into a Provider for the first time. After the first admission, every time an individual is transferred from one treatment to another the Treatment Episode dataset is used to update data. This dataset also allows the recording of individuals' final discharge information into FASAMS.

Client Specific Service Event

Description - Service Event data includes individual-level encounter data on types, amounts, locations, and dates of covered service events provided to each person served in SA or MH programs funded in whole or in part by DCF. It also includes encounter data on types, amounts, locations, and dates of covered services provided that do not require service recipients to be uniquely identified, e.g. drop-in/self-help, information and referral, and outreach.

Hierarchy - Service Event depends of Provider and Client data. Service Event data also depends of subcontract and treatment episode data based on the business rules (Conditionally required).

Waiting list

Description - Waiting List data includes information needed to identify and track individuals placed on various waiting lists for covered services. The covered services are available in community substance abuse and mental health programs or in state mental health treatment facilities. An individual is "waiting for a service" as of the date they were placed on the Waiting List and is considered "removed from the Waiting List" when an outcome is specified.

Hierarchy - Waiting List depends upon Provider and Client datasets. Waiting Lists data also depends upon subcontract and treatment episode data based on the business rules (Conditionally required).

Acute Care

Description - Acute Care Services Utilization data includes daily census of the total number of operational licensed beds, the number of these beds purchased by DCF as well as beds occupied by DCF and other payor classes. This data set also includes the distinct count of indigent individuals served daily, including the beginning census, new admissions and discharges.

Hierarchy - Acute Care dataset depends of Provider and Client data. Acute Care data also depends of subcontract data based on the business rules (Conditionally required).

Subcontract

Description - Subcontract data include specific information related to the contract between DCF's contracted entity (i.e. a Managing Entity) and the agency that provides services (i.e. a Provider).

Hierarchy - Subcontract dataset depends of Provider data only.

9.7. Requirements and Test Cases

The OZ team attempted to map the test cases to the original requirements. The team spent a week attempting to track the requirements from their original state (North Highland) up to the traceability matrix and the final test cases produced by FEI. The reviewed documents were:

1. FASAMS Requirements Specification Document and Requirements Traceability Verification Matrix (RTVM).xlsx
2. FASAMS Requirements Specification Document and Requirements Traceability Verification Matrix (RTVM) – Stage 1.xlsx
3. FASAMS Requirements Specification Document and Requirements Traceability Verification Matrix (RTVM) – Stage 2.xlsx
4. FASAMS Requirements Specification Document and Requirements Traceability Verification Matrix (RTVM) – Stage 3.xlsx
5. FASAMS Requirements Specification Document and Requirements Traceability Verification Matrix (RTVM) – Stage 4.xlsx
6. FASAMS Requirements Specification Document and Requirements Traceability Verification Matrix (RTVM) – Stage 5.xlsx
7. FASAMS Functional Design Document.docx
8. FASAMS Functional Design Document - Stage 1.docx
9. FASAMS Functional Design Document - Stage 2.docx
10. FASAMS Functional Design Document - Stage 3.docx
11. FASAMS Functional Design Document - Stage 4.docx
12. FASAMS_UAT_Results.xlsx
13. Integration and System Test Results.xlsx (the embedded test cases in this document are the same as document #12)

The team took a sample of 20 items from document #1 and document #12 above and created the document - Map Matrix-FDD-Test Cases.xlsx, which is submitted as a separate document, and the contents are also found in the Appendices, as described below. That document contains the following sheets:

- **Map matrix and FDD – See Appendix C** - Took the following columns from document #1 above to see if I could map them to documents 1-11.
    - RTVM (Document #1) - Original Requirement ID #
    - Requirement (The system will:)
    - Doc Ref 1 Reference Location / Requirement ID #

  The team had mixed results with the other versions of the RTVM (document #2-6 above) in matching the document reference numbers. Then we tried to map the values to the functional design document (document #7-11 above) with mixed results as well.

- **Map matrix to test plans - See Appendix D** – We took the following columns from document #1 above to see if we could map them to the other documents.
    - RTVM (Document #1) - Original Requirement ID #
    - Requirement (The system will:)
    - Doc Ref - Test Plan Reference Location / Test Case Identifier

  The team was not able to map most of the test plans in document #12 to the RTVM documents (#1-6).

- **Map test plan to matrix and FDD - See Appendix E** – took the test case title from document #12 and attempted to map it back to document #1 and #7. Mapping to document #1 was not very exact, but it does appear that it correlates to document #12.

After completing the tracking/mapping we assumed/concluded the following:

- The versions of the RTVM document with stages in the name (#2-6) did not appear to add any value. It seems that document #1 should be used.
- The versions of the functional design document with stages in the name (#8-11) did not appear to add any value. It seems that document #7 should be used.
- The RTVM document (document #1) has limited usefulness in mapping requirements to the functional design document (document #7).
- The UAT results (document #12) does appear to map to the functional design document (document #7).

We provided these results to DCF and asked the DCF team the following questions:

- Are the above assumptions correct?
- Are there other documents that we should be looking at to map the original requirements from North Highland to the functional design document (document #7)?
- Are there other documents that we should be looking at to map the test cases to the requirements?

We had a conference call with DCF to discuss and were provided a new document, LU915 - Amendment 002 – Executed – Redacted.pdf (Amendment 002), which was the final version of the contract signed between DCF and FEI. We spoke with the DCF resources who performed QA on the system and were told that the test cases were created directly from the Functional Design Document (#7) as they were not provided a copy of the requirements document or Amendment 002. We then attempted to map a sample of 30 requirements from Amendment 002 to the FASAMS Functional Design Document (#7). Of the 30 items, 16 were found in document #7 and 14 were not. As a result, we were not able to locate the QA test cases for all 30 items. The samples are below:

Requirement ID	Core/Sub-Process	Description	Was a match found?
00.00.0004	General	The system shall provide the capability to configure and propagate system changes (e.g., business rules, algorithms) from DCF Headquarters downstream to Managing Entities and Providers as defined by DCF.	No
00.00.0009	General	The system shall log date, time, and source when a record is added, updated or deleted.	No
00.01.0014	Architecture	The system shall be 508 compliant for ADA accessibility.	Yes
00.01.0015	Architecture	The system shall provide a user acceptance environment capable of testing all functions and interfaces used in production.	No
00.01.0017	Architecture	The system shall provide the capability to perform data quality management (e.g., data format, value, referential integrity).	Yes
00.02.0035	Data Warehouse	The system shall provide the capability to extract data from FASAMS on a nightly basis and populate a data warehouse.	No
00.05.0072	Security	The system shall be subject to FIPS 199 data integrity standards (See FIPS 199 in Supporting Documents).	No

Requirement ID	Core/Sub-Process	Description	Was a match found?
00.05.0073	Security	The system shall be subject to NIST 800-30 risk assessments (See NIST 800-30 Risk Assessment in Supporting Documents).	No
00.05.0074	Security	The system shall encrypt PII/PHI data in accordance with HIPAA.	No
00.05.0075	Security	The system shall restrict access to data within the system based on configurable FASAMS roles (see FASAMS Role Mapping in Supporting Documents).	No
00.05.0076	Security	The system shall restrict access to functionality within the system based on configurable FASAMS roles (see FASAMS Role Mapping in Supporting Documents).	Yes
00.05.0077	Security	The system shall provide the ability to perform secure file transfers using a file transfer method (e.g., SFTP, FTPS, SSH).	Yes
00.06.0054	Reporting	The system shall provide a configurable dashboard to provide summary information to authorized users with the ability to drill down.	No
00.06.0062	Reporting	The system shall provide reporting functionality for the analysis of Service Utilization (see Reports in Supporting Documents).	No
01.02.0015	Client Admission	The system shall be configurable to accept a standard formatted integrated (co-occurring) Admission record per client transaction type as defined in the FASAMS Data Model Document in compliance with TEDS requirements (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	Yes
01.02.0016	Client Admission	The system shall be configurable to accept a standard formatted demographic record per client as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	Yes
01.03.0011	Client Assessment	The system shall have the capability to calculate a standard score for level of functioning and level of care (e.g., mean, standard deviation, Z score) for a client using accepted client assessment data as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	Yes

Requirement ID	Core/Sub-Process	Description	Was a match found?
01.03.0012	Client Assessment	The system shall provide the capability to accept multiple co-occurring diagnose records per admission record and per discharge record as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	Yes
01.05.0033	Client-Specific Services	The system shall track the length of stay on a given level of care under an admission, within an episode of care.	Yes
01.06.0034	Non Client-Specific Services	The system shall be configurable to accept Non Client-Specific service data as defined in the Pamphlet 155-2 (See Pamphlet 155-2 in Supporting Documents).	No
01.06.0035	Non Client-Specific Services	The system shall provide the capability to associate Non Client-Specific service events to corresponding service contracts.	No
01.08.0036	Client Discharge	The system shall be configurable to accept a standard formatted integrated (co-occurring) Discharge record per client transaction type as defined in the FASAMS Data Model Document in compliance with TEDS requirements (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	Yes
01.08.0037	Client Discharge	The system shall provide the capability to associate discharge records to corresponding admission records.	Yes
01.09.0038	Provider Directory	The system shall be configurable to accept a standard formatted Provider record as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	Yes
02.07.0077	Invoice Validation	The system shall require each service encounter (client-specific and non-client-specific) to track the status of service event units earned that are invoiced by the provider and approved by ME for payment as part of the ME invoice validation process.	Yes
02.12.0068	Eligibility Validation	The system shall be configurable to accept client-related Medicaid eligibility data from FMMIS as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	Yes

Requirement ID	Core/Sub-Process	Description	Was a match found?
02.13.0064	Actual Expenditures	The system shall be configurable to track payment categories (e.g., Capitated, Cost Reimbursement, Bundled Rates, Capacity, Utilization) as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	No
03.14.0082	Service Reporting	The system shall provide the capability to drill down from a summary level to a client-specific level of detail as defined by DCF.	No
03.14.0083	Service Reporting	The system shall provide the capability to define standard user profiles from which individual user roles may inherit privileges.	Yes
03.14.0084	Service Reporting	The system shall provide the capability to create, activate, modify, or deactivate a user to multiple roles (see FASAMS Role Mapping in Supporting Documents).	Yes

In addition, we did not see documentation that the following "standard" QA items (or types of items) were tested as part of the overall QA process:

- **Required fields**
  - Each required field should be left blank. For example, if year of birth is required, the record with the blank date of birth shall be rejected, and the system should produce an error with a message like: "Year of birth is required".
- **Invalid data types**
  - Each field should be tested using data types that are invalid for that field. For example, year of birth will contain only numbers. When values are submitted that are non-numeric, the system shall reject the record, and system should produce an error with a message like: "ABC1 is not a valid year of birth. The value must be numeric and between 1900 and 2019."
- **Range – min/max**
  - Each field should be tested using values that are the correct data type and that are outside of the allowable range of values for that field. For example, year of birth should be between 1900 and 2019. When values are submitted that are not in that range, the system shall reject the record and produce an error with a message like: "1842 is not a valid year of birth. The value must be between 1900 and 2019."
- **Invalid values**
  - Each field should be tested using values that are the correct data type and that are not in the list of allowable values for that field. For example, if allowable gender values are "Male" and "Female", when a value of "M" is submitted, the system shall reject the record and produce an error with a message like: "'M' is not a valid gender code."
- **Dependencies**
  - Record/field should be tested where dependencies are missing. For example, if a client must be assigned to a provider, submit a record for a client attached to a provider that does not

exist. The system shall reject the record and produce an error with a message like: “Client John Smith is assigned to provider Susan Sunshine. Susan Sunshine does not exist in the system.”

- **Completeness validation**
  - Confirm that if 100 records were submitted, 20 had errors and 80 were imported correctly, that the 80 are present in the system and the 20 rejected correctly with appropriate error messages.
- **All types of records tested**
  - All types of records shall be tested – Provider, client, waiting list, subcontract, treatment episode, acute care and client specific service events.
- **Each step of the process shall be validated**
  - A file containing providers will be submitted. All records are validated – successes and errors. Then a file containing clients that receive services from providers that were just imported is submitted. This occurs with all types of files.
- **Change records for each type of file are submitted**
  - Files containing updates to existing records are submitted. The system verifies that the change is a valid one (for example, if the record is to change the provider, the provider must exist, be assigned to that ME, etc.) The system is checked to ensure that the changes were successfully saved in the system, if the record exists and the record is a valid record. Appropriate error messages are displayed and records with errors rejected by the system. The date that the record is changed is tracked. An audit trail indicating what was changed exists.
- **Deletion records for each record type are submitted**
  - Files containing records to delete existing records are submitted. The system verifies that the records marked for deletion do not have any dependent (child) records. If there are no child records, then the records should be marked deleted if all other criteria are met (record belongs to that ME, etc.). Records that have child records should not be removed and the system shall display a message like “XYZ may not be deleted because there are child records attached to it.”. Ideally, there would be a report in the system where the user can look up the records. An audit trail indicating what was deleted exists.
- **Following a record through all stages of the process**
  - As records are submitted, it should be validated as they go from upload to the staging tables, then from staging to production, and then confirm that once they are in production they appear in the reports.
- **Map all fields in the XML files to the database**
  - Verify that all fields in the files being submitted are either stored in the database or used to calculate fields that are stored in the database.

10. FASAMS Technical Review

The OZ team performed an architectural review and analyzed the technology stack used for the application. We reviewed the database – this included a detailed analysis of the structure, schemas, tables, size, indexes, volumes, and stored procedures. We also reviewed the report stored procedures and the report definitions (RDL). The OZ team did not gain access to the full source code during the course of this review, so were not able to complete the code review necessary for the full technical review. Our review revealed many serious and basic issues in the application which have severe performance and data integrity implications.

NOTE: As of this writing, the OZ team does not have access to all the source code so was not able to perform a full review of the code.

Based on our technical review, the most critical items to resolve are

1. Proper configuration of the logging framework

2. Partition tables and indexes (archiving)
3. Create a separate reporting database
4. Hardware improvements
5. Convert snowflake schema to star schema
6. Lower cardinality
7. Minimize the number of columns
8. Use less expensive data types

10.1. Purpose

The purpose of this section is to document the gaps or areas of improvement in the process or results of technical review. It also includes recommendations and conclusions that are a result of the system technical evaluation. These findings are listed based on the information provided for the FASAMS System and its technical environment. Documentation for the following components was not received so the OZ team was unable to assess them:

1. FASAMS application source code
2. Access to the Amazon cloud environment needed to review the physical architecture
3. Access to the FASAMS database

During this review process The OZ Technical Team used the FASAMS - UAT 1.1.1 document which is available on the DCF Website and on the FEI Systems SharePoint site (only that documentation for which access has been provided to the OZ team). As FASAMS is hosted on the Amazon environment and the source code was not provided and the analysis and findings of the OZ technical team depend on the provided SSRS and SSAS code, documents and database(s) (with limited permissions).

Below are the limitations which restricted the OZ technical team from performing a complete review - FASAMS source code was provided in bits and pieces, but the core or the entry point of the system's source code (ASP.Net, MVC, Angular), which integrates all the related layers in the project was not made available. The source code provided was separated from the actual solution which can be opened and viewed but cannot be compiled.

1. In order to analyze the physical structure and the different tiers of the project laid down on the Amazon cloud, the team required access to the environment. Access was not provided, however, in a meeting it was decided to spin up a UAT equivalent virtual machine for the OZ team. The environment is also needed for the hardware, network, security review.
2. Appropriate access was not provided on the different pieces of code which impacts the technical review to navigate all application areas.
3. The OZ technical team was provided limited access to the database which prevented it from doing a complete review of the database tier.
4. The OZ technical team was very much dependent on FEI Systems whenever there was a need for any environment access.

10.2. Known Performance Areas/ Issues

During the meetings and discussions with DCF and MEs, the performance areas/modules for FASAMS were highlighted, some of these known areas are listed below.

1. Reports: The users of FASAMS have repeatedly complained that reports have issues, like the performance of reports is slow and while reviewing or working with reports it was observed that the application timed out.
2. Data Submission: The Submission module in FASAMS allows users to upload file(s) and review their progress in order to see whether the process was successful or resulted in warning/errors. To fix the issues and reinitialize the data upload process, the errors report is used. FASAMS users have reported that the error identification and reporting under the submission module is not helpful in identifying the exact error and

where it occurred in the file, however currently users are doing this manually by exporting the results in Microsoft Excel.

These are the areas user(s) have pointed out. Based on these identified areas, we did a technical review of the FASAMS application to see:

1. How the reports and application architecture are designed?
2. How is the data stored?
3. Which development frameworks are used in the application?

There could be other areas which could be impacting these modules directly or indirectly which will be further detailed in the following sections.

The OZ team primarily focused on the known performance areas however tried to review other areas where access/permissions were granted.

10.3. Tools and Technologies

Due to the unavailability of FASAMS source code, we were not able to review the web application, its frontend, and its entry point. Figure 1 and Figure 2 below are taken from the documents that were shared with the OZ team which show the technology listing, however, in order to perform a review, it was required to have access to the source code which can be compiled on demand.

Software versions as listed in Figure 1 shows that FASAMS is built on Microsoft Technology Stack and using a few other open source/3rd party libraries.

We have looked into the technology stack and the software versions; the listed framework and the software versions are 2016/2017 including the Windows operating system where the FASAMS Application(s)/Database(s) are hosted.

As per the session with FEI Systems Inc., the OZ team was told that the database and application servers for FASAMS are hosted on a Virtual Machine (VM) on the Amazon (AWS) cloud. The VM is running a Windows OS which contains the UAT and Production.

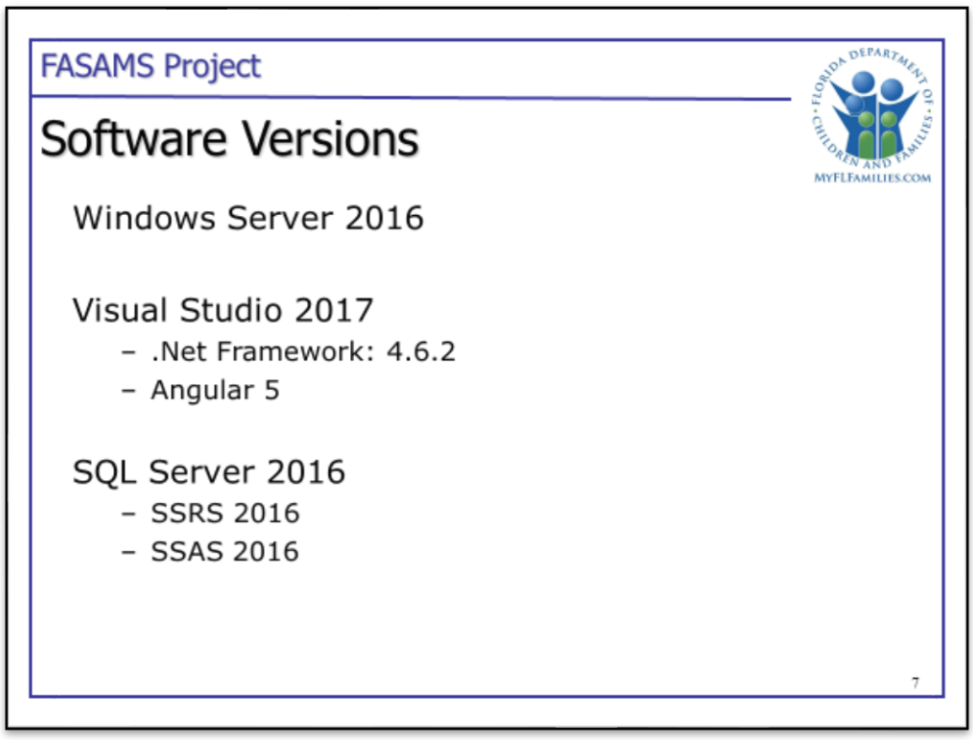


FIGURE 83 - SOFTWARE VERSIONS

After reviewing the list of open source/ 3rd party libraries we have found that these libraries are being used for UX, Data, Security and for logging. These libraries are listed below:

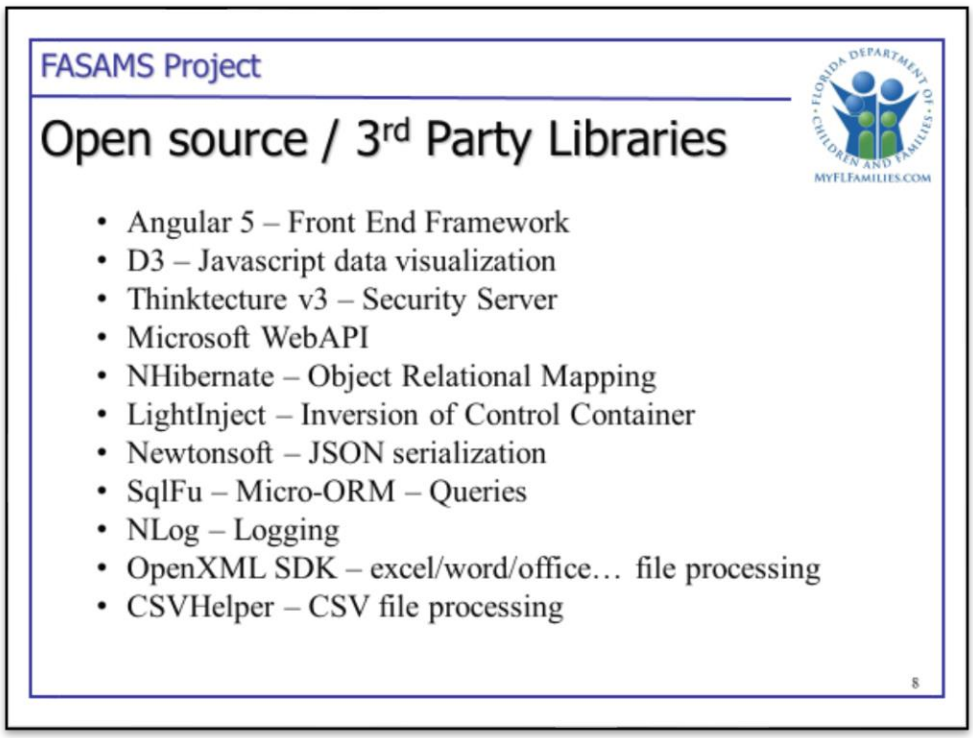


FIGURE 84 - OPEN SOURCE / 3RD PARTY LIBRARIES

Since the OZ team was not provided access to the Web Application/Front End/Entry Point for the FASAMS, we can only assume what these libraries could have been used for inside the different layers of the project by looking at the individual library features/functionality.

Angular 5 is used as a frontend framework.

D3 is a JavaScript library for visualizing data with HTML, SVG, and CSS. However, we couldn't find any web pages in FASAMS visualizing data with HTML.

IdentityServer is a framework and a hostable component that allows implementing single sign-on and access control for modern web applications and APIs using protocols like OpenID Connect and OAuth2. It supports a wide range of clients like mobile, web, SPAs and desktop applications and is extensible to allow integration in new and existing architectures. IdentityServer is a project of the independent Thinktecture Associates. IdentityServer -- the Open Source OpenID Connect and OAuth 2.0 framework for .NET. IdentityServer is OpenID Certified and part of the .NET Foundation.

Mictososft WebAPI is used which is the secure REST APIs that reach a broad range of clients including browsers and mobile devices.

NHibernate is used as the object relational mapping (ORM) solution for .Net Framework.

LightInject is used which is the ultra-lightweight Inversion of Control (IoC) Container.

Newtonsoft JSON serialization is used which is to convert between JSON text and .Net objects.

SqlFu is Fast, versatile Ado.Net data mapper with helpers to improve productivity when working with a RDBMS. It is designed for flexibility and ease of use.

NLog is a flexible and free logging platform for various .NET platforms, including .NET standard. NLog makes it easy to write to several targets (database, file, console) and change the logging configuration on-the-fly. NLog has support for structured and traditional logging.

The SDK is built on the System.IO.Packaging API and provides strongly-typed classes to manipulate documents that adhere to the Office Open XML File Formats specification. The Office Open XML File Formats specification is an open, international, ECMA-376, Second Edition and ISO/IEC 29500 standard.

A .NET library for reading and writing CSV files. Extremely fast, flexible, and easy to use.

While analyzing the FASAMS Web Application we saw that Qill.js is being used. Qill.js is a free, open source WYSIWYG editor built for the modern web. With its modular architecture and expressive API, it is completely customizable to fit any need.

NHibernate and Entity Framework both (see Table 1 below) are equally capable Object Relation Mapping (ORM) and suitable for all types of ASP.NET software applications. In view of ASP.NET software companies, both frameworks are used for processing relational data to domain specific objects. Entity framework is easier for someone familiar with Microsoft stack technology.

Both NHibernate and the Entity Framework provide the capabilities required to implement an ORM using the Model First approach. The Framework creates the classes defined in the data model, NHibernate does not. Nonetheless, from a developer point of view, creating the entities and their relationships by hand does provide greater insight into the detailed aspects of the program.

TABLE 1 - COMPARISON BETWEEN NHIBERNATE VS. ENTITY FRAMEWORK

Properties	NHibernate	Entity Framework
Mapping	It supports XML-based, attribute-based and strongly-type code-based mappings	It supports attribute-based and strongly-typed code mappings
Database Support	All types of DB providers are supported e.g. Microsoft SQL Server (including Compact Edition), Oracle, Microsoft Access, etc.	It supports SQL Server, SQL server compact edition provider, SQL Lite, SQL Azure, Oracle, and MySQL
Code First Mapping Support	Yes	Introduced from version 4.0
Lazy Loading Support	Lazy loading for Associated Entities (one to one, many to one) Collections (one to many, many to many); Scalar properties (thing of BLOBs or CLOBs)	Lazy loading for Associated Entities and Collections
Migrations Support	Supports only initial schema generation	Built-in schema migration support, also supports seeding of the initial database
Asynchronous call	No	Yes
Connection resiliency	Third-party solutions exist with NHibernate.SqlAzure	This feature was added to version 6; the connection resiliency feature automates the process of retrying failed SQL statements
Code based Configuration	NHibernate 3.2 with Fluent-NHibernate	Available from version 4.1. Entity Framework automatically finds a class that derives from DbConfiguration. One can use the DbConfiguration class to complete configuration tasks in the source code
Querying API	LINQ provider for NHibernate and Query Over. This is similar to Criteria API but uses strongly-typed LINQ expressions instead of strings. This is also the most commonly used today	LINQ to Entities most used Entity-SQL-object oriented, database independent querying language Plain SQL
Documentation	Poor, but good forum support is available	Good
Cascading Support	Yes	Yes
Caching	Support second level caching	Yes, from version 6.0
Cascading Support	Yes	Yes

Properties	NHibernate	Entity Framework
Tracking changes	Change tracking at the unit of work level	Change tracking at the unit of work level. Entity Framework also offers self-tracking entities.
Events	NHibernate has a very rich event model, that exposes more than 20 events, either for synchronous pre-execution or asynchronous post-execution, including: Pre/Post-Load, Pre/Post-Delete, Pre/Post-Insert, Pre/Post-Update, and Pre/Post-Flush.	Entity Framework only has two event-based extension points: ObjectMaterialized and SavingChanges.
Batching Support	NHibernate has full support for insertion batching, but only if the ID generator in use is not database-based (for example, it cannot be used with Identity)	No
Flushing Changes	NHibernate's ISession has a FlushMode property that can have the following values: • Auto • Commit • Never	Entity Framework, changes have to be explicitly sent through a call to AcceptAllChanges() and SaveChanges()
Custom type and collection support	Yes	No
.NET 4.6 Support	Yes	Yes (with improved performance)
Complexity	Higher	Easy

As it is stated above that NLog is a logging framework used in .Net Framework and can write to several targets i.e. database, file, and console. We have found FASAMS uses a database target as the log is being written to a table which is explained in more detail in section 6.4 Database.

10.4. Database

In this section we have outlined the high-level review for the FASAMS database(s) where, based on the limited permissions granted, The OZ team has looked into following four (4) databases to perform a high-level technical review.

10.4.1. Bird.Portal.Database.Fasams.Uat

TABLE 2 - SCHEMA AND TABLES COUNT

Schema	Table Count	Schema	Table Count
AcuteCareModule	4	MedicaidModule	9

Schema	Table Count	Schema	Table Count
BudgetModule	2	MentalHealthClientExtractModule	4
ClientModule	1	ProviderModule	14
ConfigurationModule	2	SecurityModule	9
ContractModule	6	ServiceEventModule	4
CustomerModule	5	SsasReportingModule	3
DataWarehouseModule	2	StagingModule	43
DomainHistoryModule	2	SubcontractModule	4
DynamicDataSet_FAADAModule	2	SubmissionModule	8
DynamicDataSet_FITTModule	2	TaskSchedulerModule	3
DynamicDataSet_TestModule	2	TedsExtractModule	4
DynamicDataSet_TestUATAFModule	2	TreatmentEpisodeModule	18
DynamicDataSetModule	6	VocabularyModule	16
LicensingModule	1	WaitingListModule	1
LoggingModule	3	X12Module	3

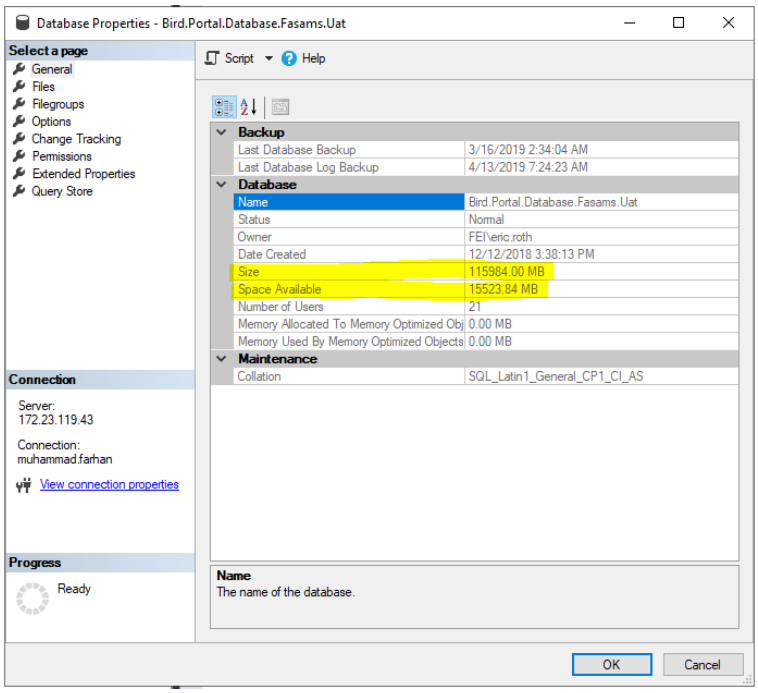


FIGURE 85 - DATABASE SIZE

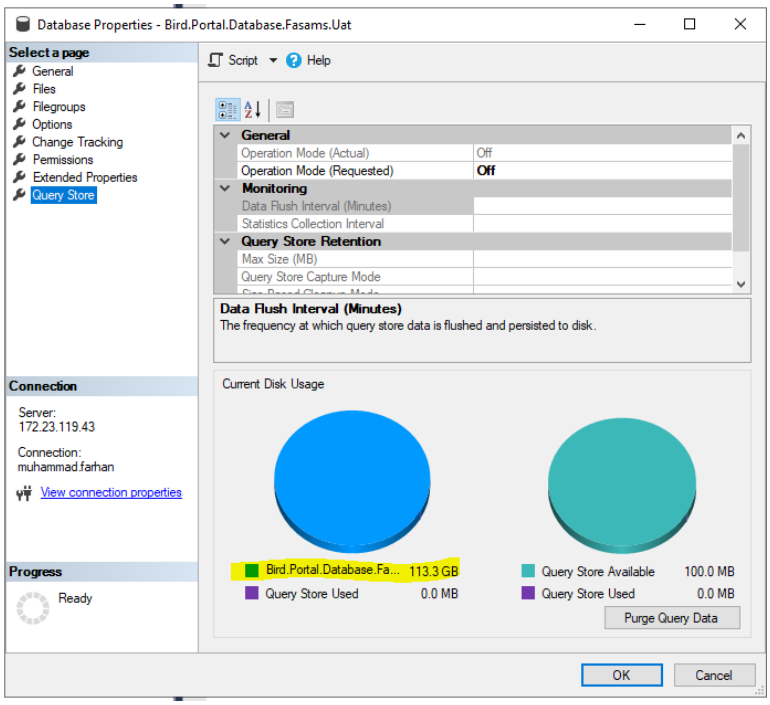


FIGURE 86 - DATABASE QUERY STORE

10.4.2. Bird.Ssrs.Database.Fasams.Uat

TABLE 3 - SCHEMA AND TABLES COUNT

Schema	Table Count
dbo	36

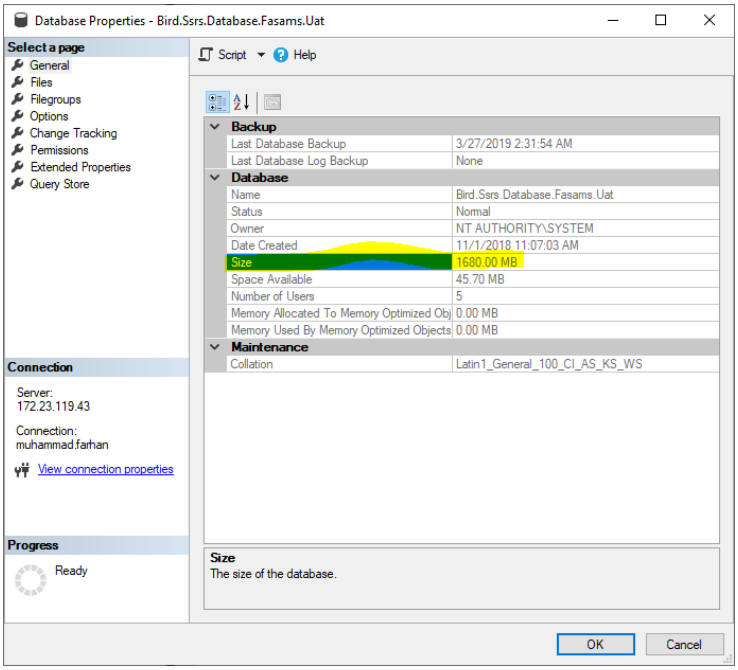


FIGURE 87 - DATABASE SIZE

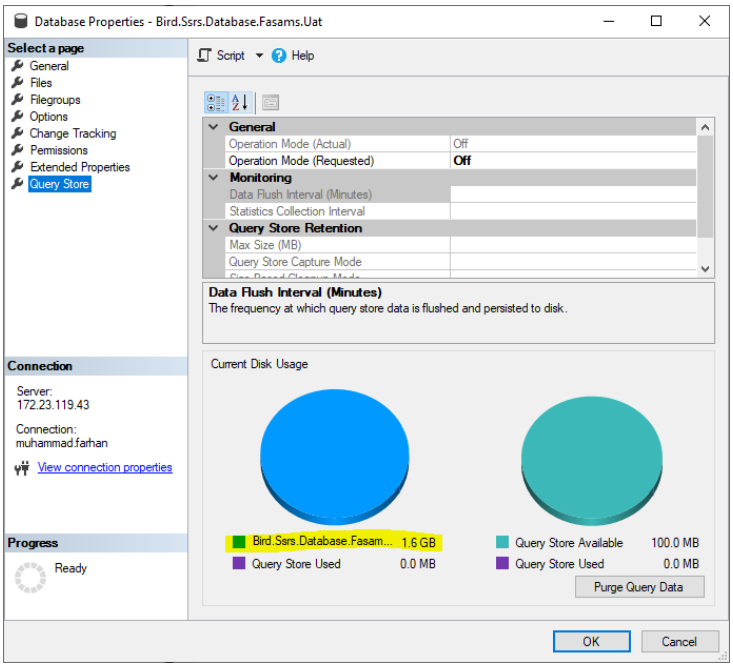


FIGURE 87 - DATABASE QUERY STORE

10.4.3. Bird.Ssrs.Database.Fasams.UatTempDB

TABLE 4 - SCHEMA AND TABLES COUNT

Schema	Table Count
dbo	14

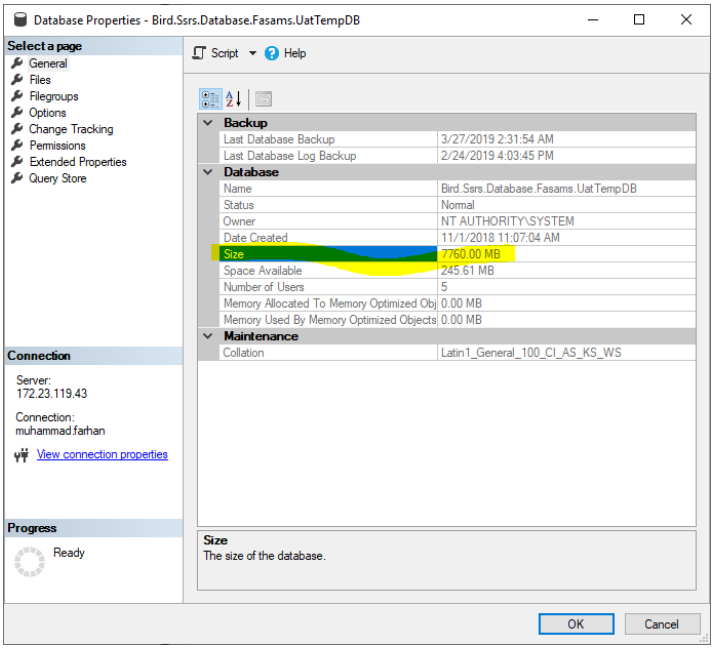


FIGURE 88 - DATABASE SIZE

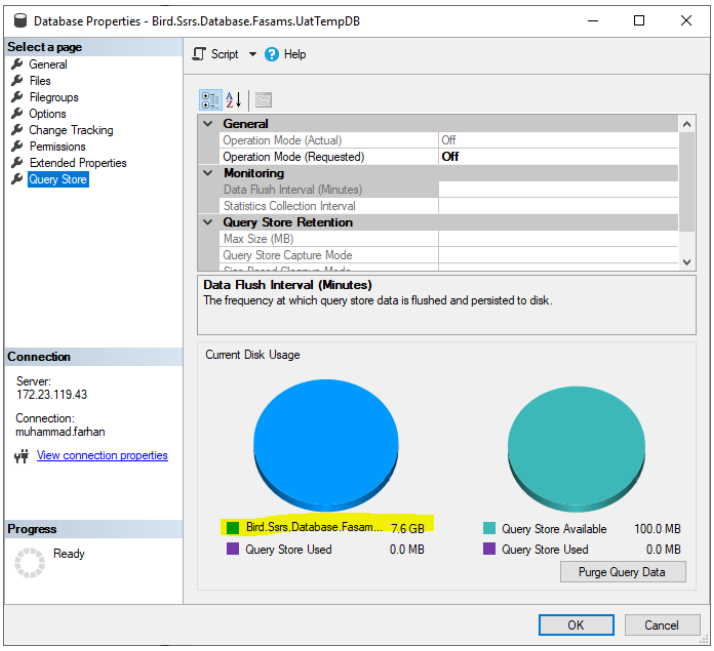


FIGURE 89 - DATABASE QUERY STORE

10.4.4. Pillar.SecurityServer.Database.Fasams.Uat

TABLE 5 - SCHEMA AND TABLES COUNT

Schema	Table Count
SecurityModule	29

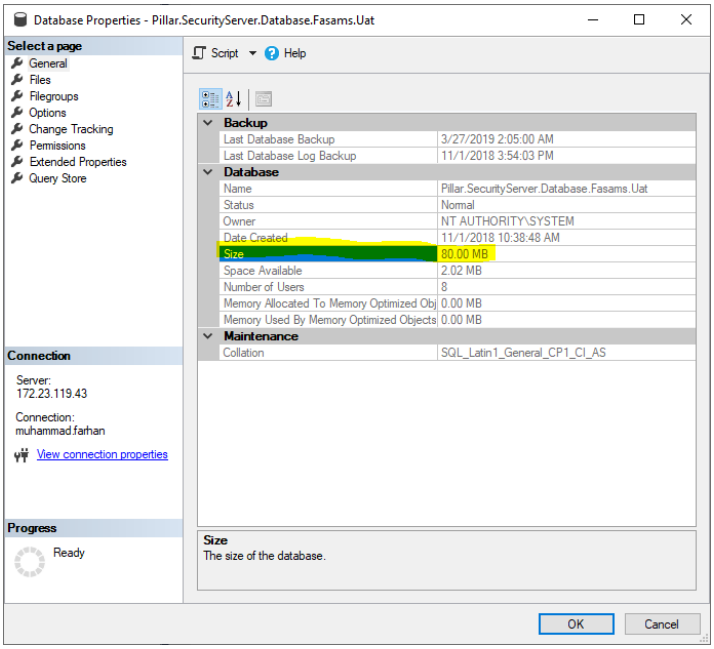


FIGURE 90 - DATABASE SIZE

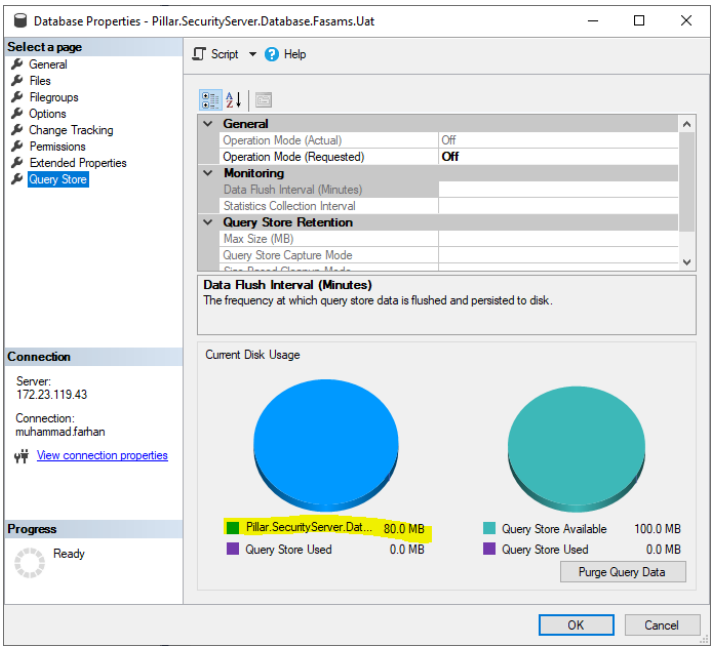


FIGURE 91- DATABASE QUERY STORE

Bird.Portal.Database.Fasams.Uat database is the largest in size and contains all the transaction data being uploaded in FASAMS.

We have looked into the Bird.Portal.Database.Fasams.Uat database and found that there is a large volume of data in two (2) tables as shown in section 6.5 - however one of these tables is being used or storing data for logging.

1. [LoggingModule].[PerformanceLog]
2. [DataWarehouseModule].[StagingSummary]

We have further investigated the [LoggingModule].[PerformanceLog] table where it is storing the log information in database (such as trace log) and by reading the data stored.

The following screenshot shows a simple query which was executed on the log table, after 4 minutes it was still running and had not completed its execution.

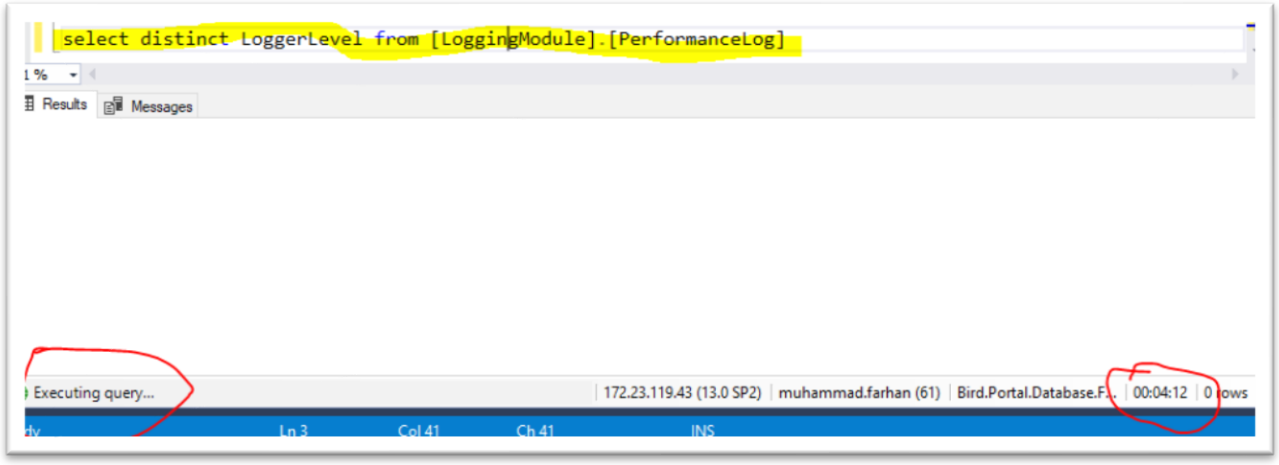
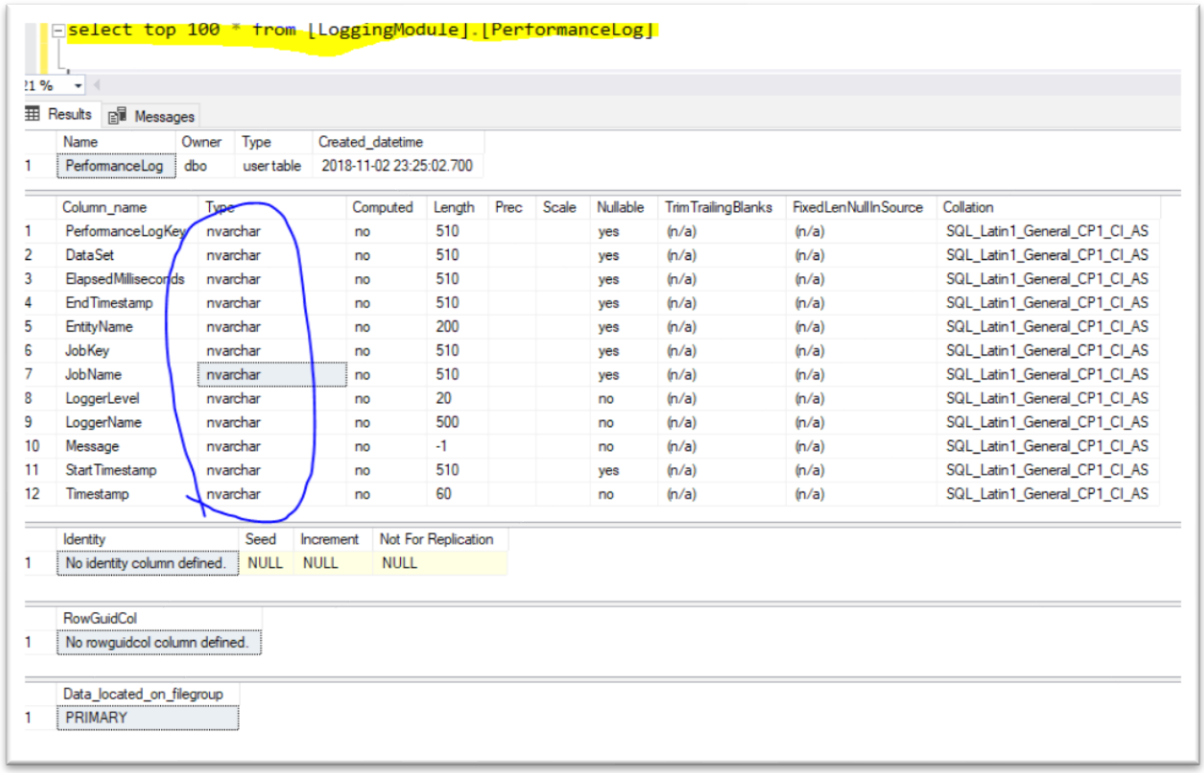


FIGURE 92 - SIMPLE QUERY EXECUTION RESULT

Further looking into the design of the table, we found that all columns have NVARCHAR data type whereas, in the data view, we see TimeStamp, GUID, and other kind of values are stored in the columns.



```
select top 100 * from [LoggingModule].[PerformanceLog]
```

Results | Messages

	Name	Owner	Type	Created_datetime
1	PerformanceLog	dbo	user table	2018-11-02 23:25:02.700

	Column_name	Type	Computed	Length	Prec	Scale	Nullable	TrimTrailingBlanks	FixedLenNullInSource	Collation
1	PerformanceLogKey	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
2	DataSet	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
3	ElapsedMilliseconds	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
4	EndTimestamp	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
5	EntityName	nvarchar	no	200			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
6	JobKey	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
7	JobName	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
8	LoggerLevel	nvarchar	no	20			no	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
9	LoggerName	nvarchar	no	500			no	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
10	Message	nvarchar	no	-1			no	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
11	StartTimestamp	nvarchar	no	510			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
12	Timestamp	nvarchar	no	60			no	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS

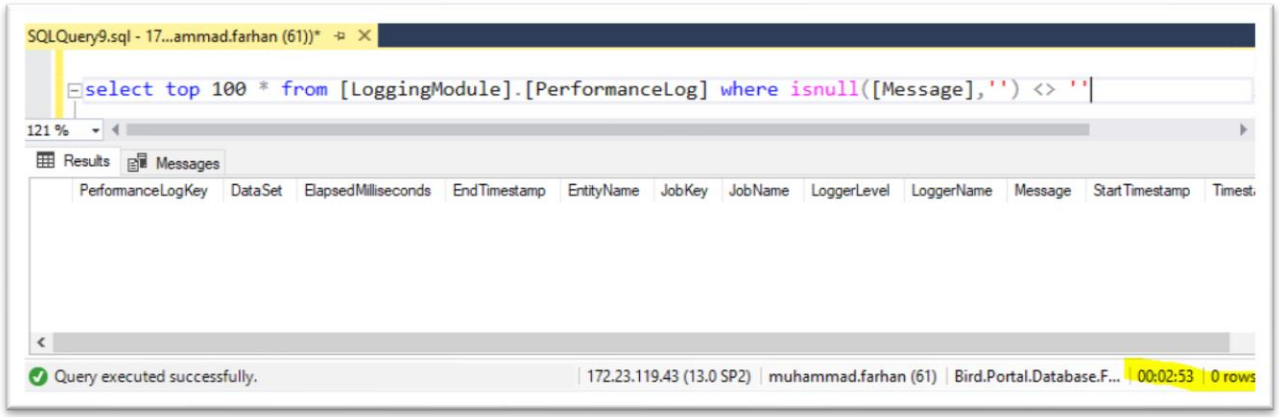
	Identity	Seed	Increment	Not For Replication
1	No identity column defined.	NULL	NULL	NULL

	RowGuidCol
1	No rowguidcol column defined.

	Data_located_on_filegroup
1	PRIMARY

FIGURE 93 - PERFORMANCE LOG TABLE DESIGN

The table contains over 29 million rows with a size of over 30GB and does not contain any data in the message column, so essentially it is a large table that contains no useful information. Furthermore, the maximum length of the message column is -1, which further begs the question – why does this table even exist in the database?



SQLQuery9.sql - 17...ammad.farhan (61))*

```
select top 100 * from [LoggingModule].[PerformanceLog] where isnull([Message],'') <> ''
```

121 %

Results | Messages

PerformanceLogKey	DataSet	ElapsedMilliseconds	EndTimestamp	EntityName	JobKey	JobName	LoggerLevel	LoggerName	Message	StartTimestamp	Timest

Query executed successfully. | 172.23.119.43 (13.0 SP2) | muhammad.farhan (61) | Bird.Portal.Database.F... | 00:02:53 | 0 rows

FIGURE 94 - QUERY SHOWING NO MESSAGE RESULTS

Also, it has been found that this table is only using Trace level logging as shown in Figure 6 below.

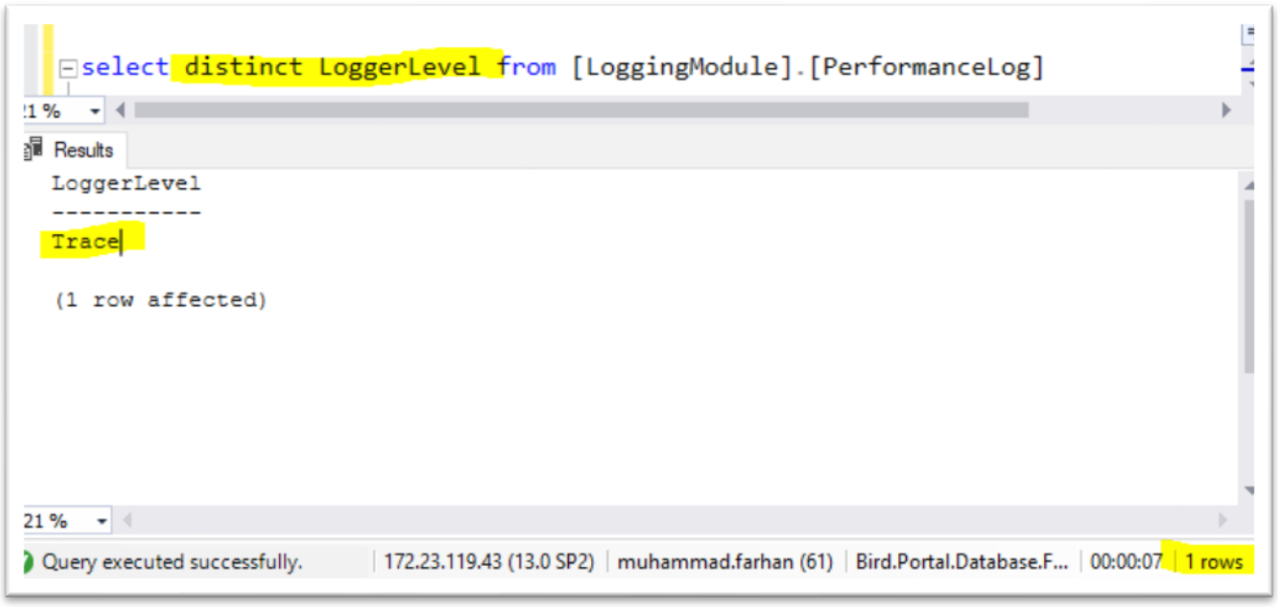


FIGURE 95 - QUERY SHOWING LOGGERLEVEL IN DATABASE

Several Log Levels could be logged based on the nature of the issue or logging of the system.

1. TRACE – informational messages that are most useful to debug an application.
2. DEBUG – fine grained informational events that are most useful to debug an application.
3. INFO – informational messages that highlight the progress of the application at coarse-grained level.
4. WARN – potentially harmful situation.
5. ERROR – error events that might still allow the application to continue running.
6. FATAL – very sever error events that will presumably lead the application to abort.

The following screenshots show the data which is stored in the table:

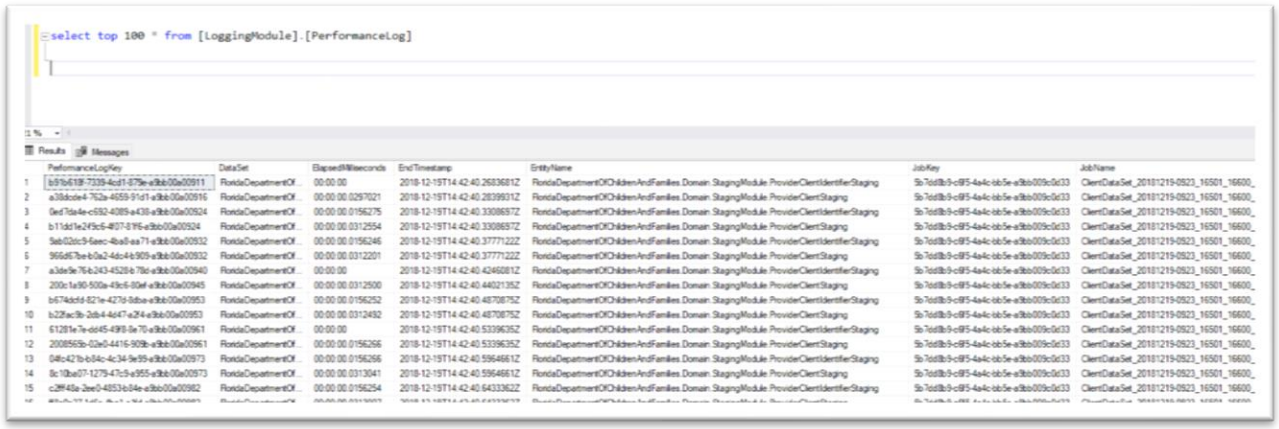
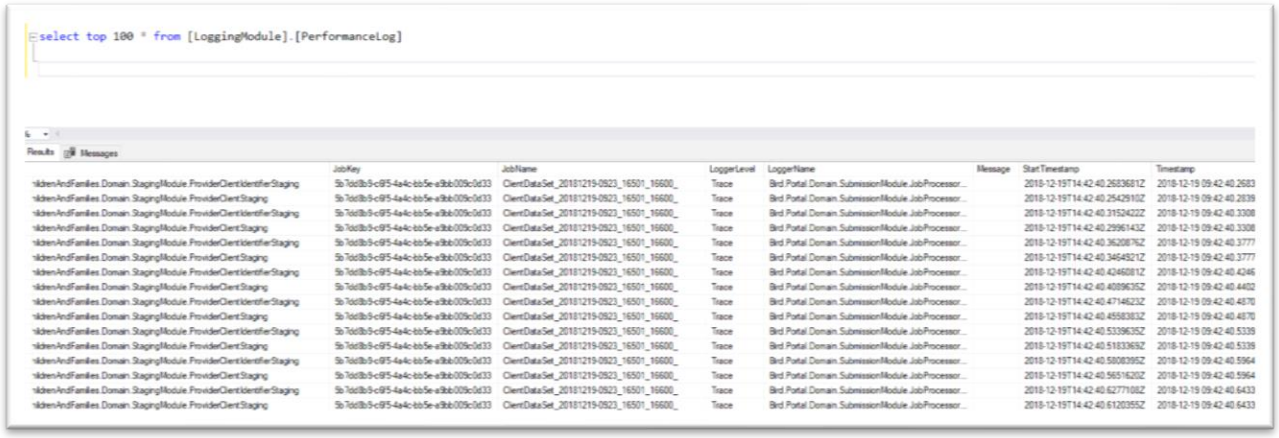


FIGURE 96 - DATA IN PERFORMANCELOG TABLE

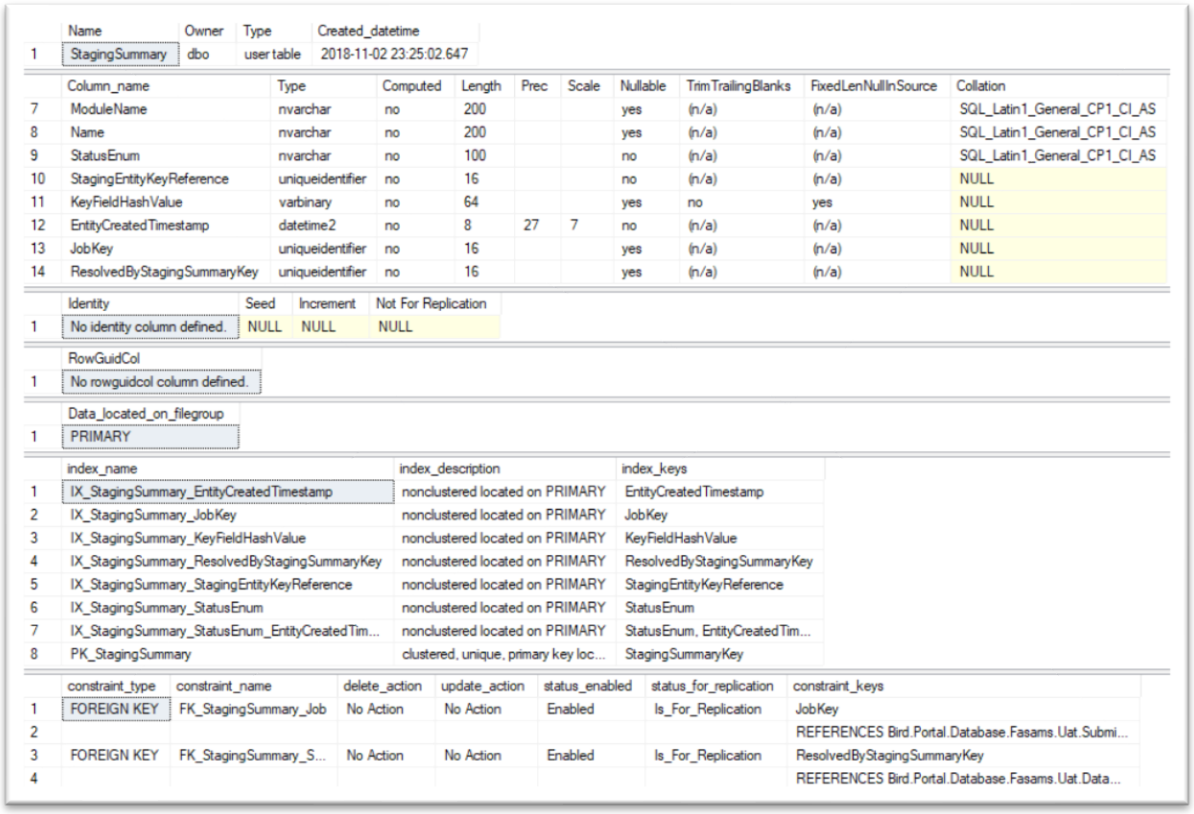


```
select top 100 * from [LoggingModule].[PerformanceLog]
```

JobKey	JobName	LoggerLevel	LoggerName	Message	StartTimestamp	Timestamp
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.2683681Z	2018-12-19 09:42:40.2683
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.2542910Z	2018-12-19 09:42:40.2839
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.3152422Z	2018-12-19 09:42:40.3308
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.2996143Z	2018-12-19 09:42:40.3308
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.3620876Z	2018-12-19 09:42:40.3777
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.3464521Z	2018-12-19 09:42:40.3777
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.4246081Z	2018-12-19 09:42:40.4246
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.4089635Z	2018-12-19 09:42:40.4402
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.4714623Z	2018-12-19 09:42:40.4870
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.4558383Z	2018-12-19 09:42:40.4870
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.5339635Z	2018-12-19 09:42:40.5339
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.5183365Z	2018-12-19 09:42:40.5339
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.5808395Z	2018-12-19 09:42:40.5964
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.5651620Z	2018-12-19 09:42:40.5964
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.6277108Z	2018-12-19 09:42:40.6433
5b7dd3b9-c9f5-4a4c-bb5e-a9bb009c0d33	ClientDataSet_20181219-0923_16501_16600_	Trace	Bird.Portal.Domain.SubmissionModule.JobProcessor...		2018-12-19T14:42:40.6120355Z	2018-12-19 09:42:40.6433

FIGURE 97 - DATA IN PERFORMANCELOG TABLE

Additionally, we started looking into the [DataWarehouseModule].[StagingSummary] table which is the 2nd largest table in the database as mentioned below (6.4.5 Data Volumes, Load Structure and Frequency). Below is a screenshot showing the table design. The Staging Summary table is also being used in most of the reports. This is the 2nd largest table in the database (28 million rows, 23GB) and it will keep growing with the passage of time (almost 16% annually) and the usage of the application. When we looked into the reports and executed the queries that are using this table, it completed very slowly - this is directly impacting the performance of reports.



	Name	Owner	Type	Created_datetime
1	StagingSummary	dbo	user table	2018-11-02 23:25:02.647

	Column_name	Type	Computed	Length	Prec	Scale	Nullable	TrimTrailingBlanks	FixedLenNullInSource	Collation
7	ModuleName	nvarchar	no	200			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
8	Name	nvarchar	no	200			yes	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
9	StatusEnum	nvarchar	no	100			no	(n/a)	(n/a)	SQL_Latin1_General_CP1_CI_AS
10	StagingEntityKeyReference	uniqueidentifier	no	16			no	(n/a)	(n/a)	NULL
11	KeyFieldHashValue	varbinary	no	64			yes	no	yes	NULL
12	EntityCreatedTimestamp	datetime2	no	8	27	7	no	(n/a)	(n/a)	NULL
13	JobKey	uniqueidentifier	no	16			yes	(n/a)	(n/a)	NULL
14	ResolvedByStagingSummaryKey	uniqueidentifier	no	16			yes	(n/a)	(n/a)	NULL

	Identity	Seed	Increment	Not For Replication
1	No identity column defined.	NULL	NULL	NULL

	RowGuidCol
1	No rowguidcol column defined.

	Data_located_on_filegroup
1	PRIMARY

	index_name	index_description	index_keys
1	IX_StagingSummary_EntityCreatedTimestamp	nonclustered located on PRIMARY	EntityCreatedTimestamp
2	IX_StagingSummary_JobKey	nonclustered located on PRIMARY	JobKey
3	IX_StagingSummary_KeyFieldHashValue	nonclustered located on PRIMARY	KeyFieldHashValue
4	IX_StagingSummary_ResolvedByStagingSummaryKey	nonclustered located on PRIMARY	ResolvedByStagingSummaryKey
5	IX_StagingSummary_StagingEntityKeyReference	nonclustered located on PRIMARY	StagingEntityKeyReference
6	IX_StagingSummary_StatusEnum	nonclustered located on PRIMARY	StatusEnum
7	IX_StagingSummary_StatusEnum_EntityCreatedTim...	nonclustered located on PRIMARY	StatusEnum, EntityCreatedTim...
8	PK_StagingSummary	clustered, unique, primary key loc...	StagingSummaryKey

	constraint_type	constraint_name	delete_action	update_action	status_enabled	status_for_replication	constraint_keys
1	FOREIGN KEY	FK_StagingSummary_Job	No Action	No Action	Enabled	Is_For_Replication	JobKey
2							REFERENCES Bird.Portal.Database.Fasams.Uat.Submi...
3	FOREIGN KEY	FK_StagingSummary_S...	No Action	No Action	Enabled	Is_For_Replication	ResolvedByStagingSummaryKey
4							REFERENCES Bird.Portal.Database.Fasams.Uat.Data...

FIGURE 98 - STAGING SUMMARY TABLE DESIGN

Staging Summary Table Dependencies lists the objects which are dependent on this table and all of them are related to SSRS as the object name suggests.

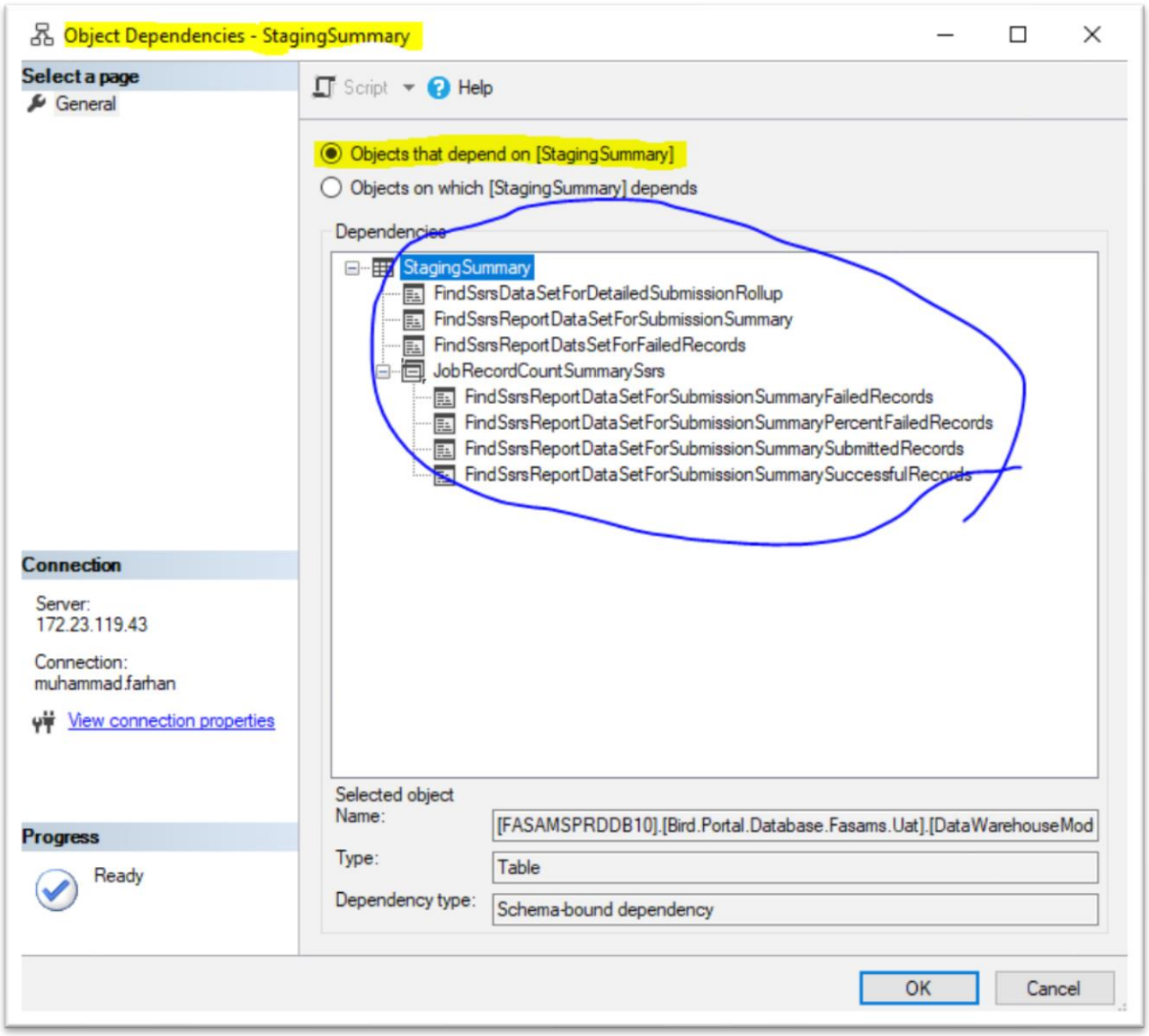


FIGURE 99 - STAGINGSUMMARY TABLE DEPENDENCIES

After reviewing these two tables we have identified the following gaps or questions on the table/database design:

1. As the database size grows with the passage of time and the usage of the system, application and report performance will be adversely impacted. Even a simple operation involving these tables could become very costly.
2. The question arises when there is large volume (currently 29GB) of logging data, why is it being stored in the database? Also, the Message field is empty in the entire logging table and only trace level information being stored.
3. Is there any table partitioning strategy in place as the growth in data will continue to slow down the application performance?
4. Is there any archiving or purge policy in place for the large tables?

5. We observed that these are normalized database(s) which is another factor for the poor performance of reports because even a simple report may require several joins for retrieving the data to be included in the report.
6. The current database also includes staging objects whereas these should not be part of the transactional database. Staging objects should have their own separate database.

10.4.5. Data Volumes, Load Structure and Frequency

Below are the top 10 tables from the Bird.Portal.Database.Fasams.Uat database to show the data volumne, used space and index size however The OZ team does not have permission to select the Fragmentation size from the database

TABLE 6 - TOP 10 HIGH VOLUME TABLES

#	Table Name	Row Count	Used Space (MB)	Index Size (KB)	Fragmentation (%)
1	[LoggingModule].[PerformanceLog]	29,072,742	30,178.03	224	The OZ team needs access to answer this.
2	[DataWarehouseModule].[StagingSummary]	28,959,665	23,206.48	16769000	
3	[StagingModule].[ServiceEventStaging]	4,251,407	3,831.71	642072	
4	[SubmissionModule].[JobError]	4,141,979	6,791.30	469416	
5	[StagingModule].[ProviderClientStaging]	2,021,972	1,335.50	307136	
6	[StagingModule].[ProviderClientIdentifierStaging]	2,013,668	1,213.16	385512	
7	[ServiceEventModule].[ServiceEvent]	1,642,216	5,346.80	2961984	
8	[StagingModule].[PerformanceOutcomeMeasureStaging]	1,443,764	1,820.55	851304	
9	[StagingModule].[StabilityOfHousingStaging]	1,443,676	844.49	219392	
10	[StagingModule].[FinancialAndHouseholdStaging]	1,443,522	917.95	219432	

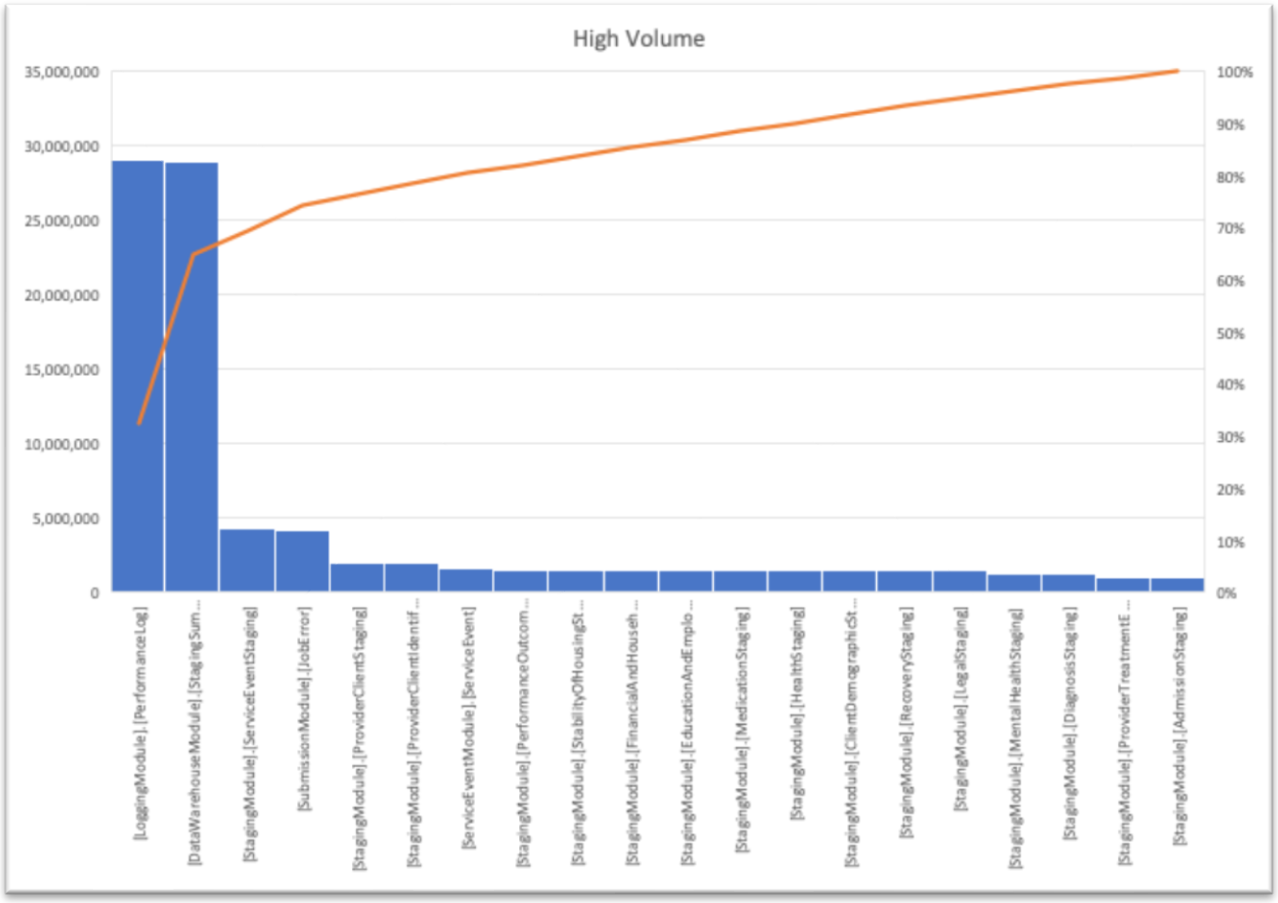


FIGURE 100 - HIGH VOLUME DATA TABLES

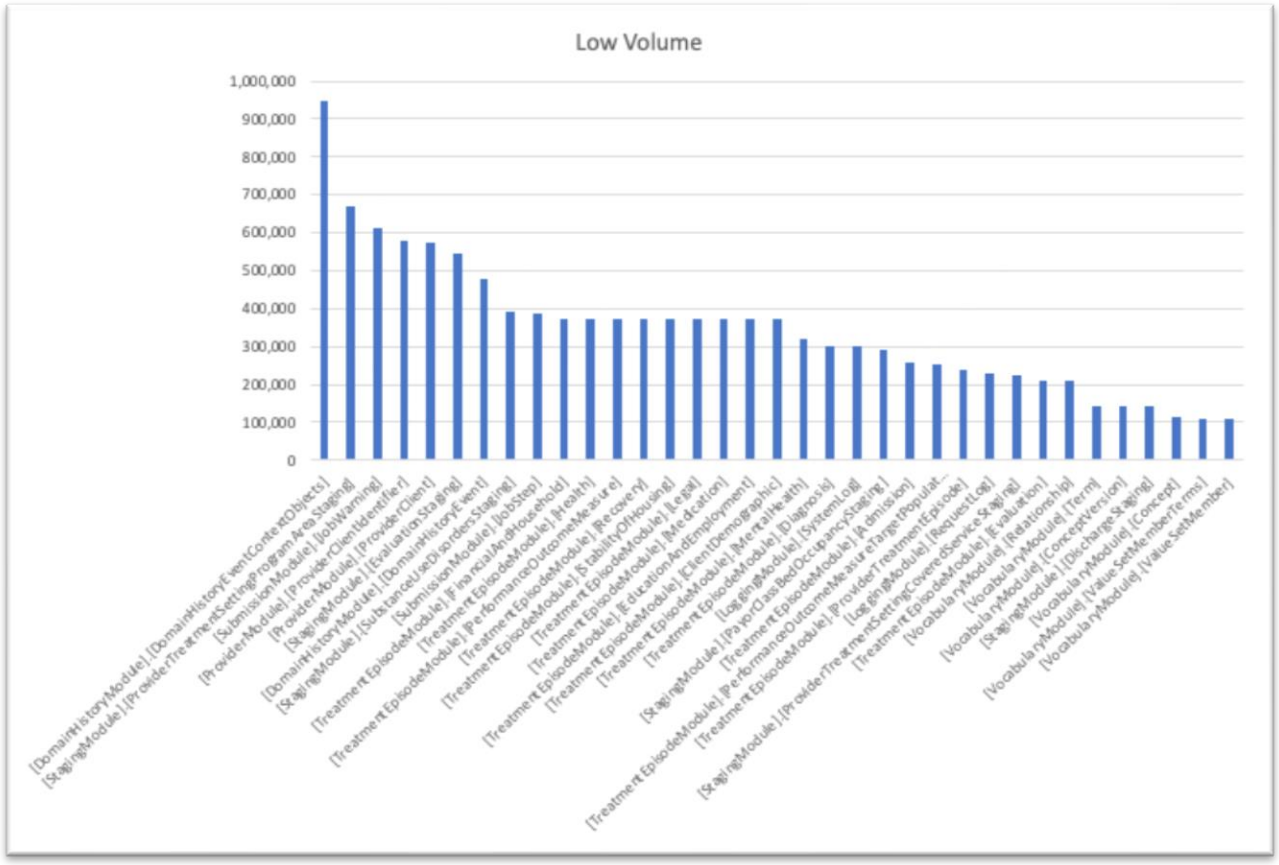


FIGURE 101- LOW VOLUME DATA TABLES

10.4.6. Database Recommendations

Partition the tables and indexes

We recommend partitioning the tables and indexes. SQL Server supports table and index partitioning. The data of partitioned tables and indexes is divided into units that can be spread across more than one filegroup in a database. The data is partitioned horizontally, so that groups of rows are mapped into individual partitions. All partitions of a single index or table must reside in the same database. The table or index is treated as a single logical entity when queries or updates are performed on the data

Benefits of Partitioning

Partitioning large tables or indexes can have the following manageability and performance benefits.

1 You can transfer or access subsets of data quickly and efficiently, while maintaining the integrity of a data collection. For example, an operation such as loading data from an OLTP to an OLAP system takes only seconds, instead of the minutes and hours the operation takes when the data is not partitioned.
2 You can perform maintenance operations on one or more partitions more quickly. The operations are more efficient because they target only these data subsets, instead of the whole table. For example, you can choose to compress data in one or more partitions or rebuild one or more partitions of an index.
3 You may improve query performance, based on the types of queries you frequently run and on your hardware configuration. For example, the query optimizer can process equijoin queries between two or more partitioned tables faster when the partitioning columns are the same as the columns on which the tables are joined.

Optimize Database indexes for better SQL performance (SSRS and Application)

An index is an on-disk structure associated with a table or view that speeds retrieval of rows from the table or view. indexes are paramount to achieving good database and application performance. Poorly designed indexes and a lack of the same are primary sources of poor SQL Server performance. The following should be considered to improve the SSAS and SSRS performance.

- Review existing Clustered and Non-Clustered Indexes
- Identify and remove index fragmentation
- Identify skewed and outdated indexes and column statistics and ensure they are representative and up to date
- Identify and create missing indexes
- Identify and remove unused indexes
- Creating and monitoring index maintenance jobs
- Consider using columnstore indexes

Columnstore indexes are the standard for storing and querying large data warehousing fact tables. Use the columnstore index to achieve up to **10x query performance** gains over traditional row-oriented storage, and up to **7x data compression** over the uncompressed data size.

A column store index is an index that was designed mainly for improving the query performance for workloads with very large amounts of data (e.g. data warehouse fact tables). This type of index stores the index data in a column-based format rather than row-based as is done with traditional indexes. There are two main benefits of column store indexes. First, they reduce storage costs. Column store indexes provide a very high level of compression, up to 10x, due to the fact that the data across columns is usually very similar and will compress quite well. Second is better performance. This benefit is multi-faceted. With a smaller index footprint, due to the compression, we reduce the amount of IO we have to perform. Also because of this small footprint we can fit more of the index into memory which helps to speed up processing

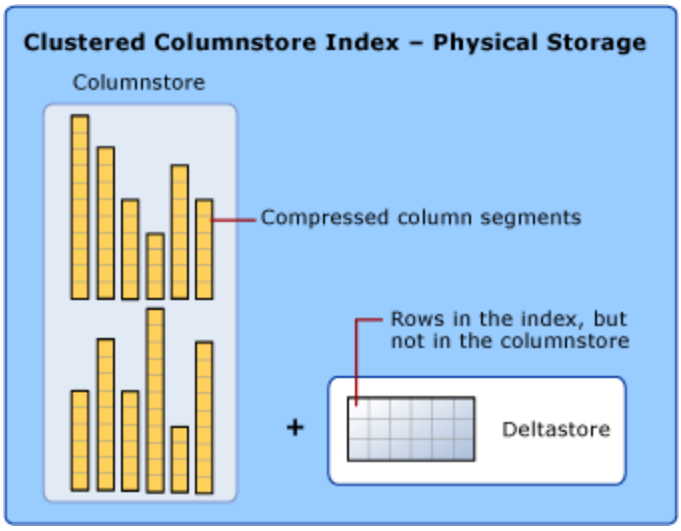


FIGURE 102 - CLUSTERED COLUMNSTORE INDEX

Data Archiving Process

Data archiving is the process of moving data that is no longer actively used to a separate storage device for long-term retention. Archive data consists of older data that remains important to the organization or must be retained

for future reference or regulatory compliance reasons. Data archives are indexed and have search capabilities, so files can be located and retrieved.

The greatest benefit of archiving data is it reduces the cost of primary storage. Primary storage is typically expensive, because a storage array must produce a sufficient level of IOPS to meet operational requirements for user read/write activity. In contrast, archive storage costs less, because it is typically based on a low-performance, high-capacity storage medium.

Archive storage also reduces the volume of data that must be backed up. Removing infrequently accessed data from the backup data set improves backup and restore performance.

10.5. Reports

During the meetings with DCF on Data and Reporting, it was communicated that the performance of reports is one of the most impacted area in the FASAMS application which needs to be addressed. There were several issues mentioned during the session which were: the reports load time was slow, the application times out while reviewing the reports, and the ad-hoc reports creation process is not user friendly where they gave an example that to write a simple query includes 4 to 5 tables in a join in order to get the intended results.

The OZ Team was given the source code for the reports project “Bird.Portal.Reports” which contains the RDL files which are used for the application reports and UAT Database(s). The OZ team reviewed the different components of reports like FASAMS application Report section, Reports Source Code and the Backend solution which includes the Stored Procedures and SSAS tabular model.

FASAMS Reports are developed in SQL Server 2016 as also mentioned in the TechnicalArchitecture.pptx [2] document. Below are the tools and technologies being used for the reports:

- SQL Server Reporting Services (SSRS)
- SQL Server Analysis Service (SSAS)

10.5.1. SQL Server Reporting Services (SSRS)

The FASAMS administrative portal allows users to run SSRS reports based on data contained within FASAMS. Below are the user roles that have access to the FASAMS reporting module:

- DCF Administrators
- Regional Office Administrators
- Regional Office Staff
- Submitting Entity Administrators
- Submitting Entity Staff

Further ad hoc report development functionality is available for DCF users with the appropriate permissions. These features are supported by Microsoft SQL Server Report Services (SSRS) and Report Builder. FASAMS published a Microsoft SQL Server Analysis Services (SSAS) tabular reporting model, which exposed the entities in the FASAMS ODS database.

Below are the findings for some of the reports:

1. Job Submission Performance ->Failed Record

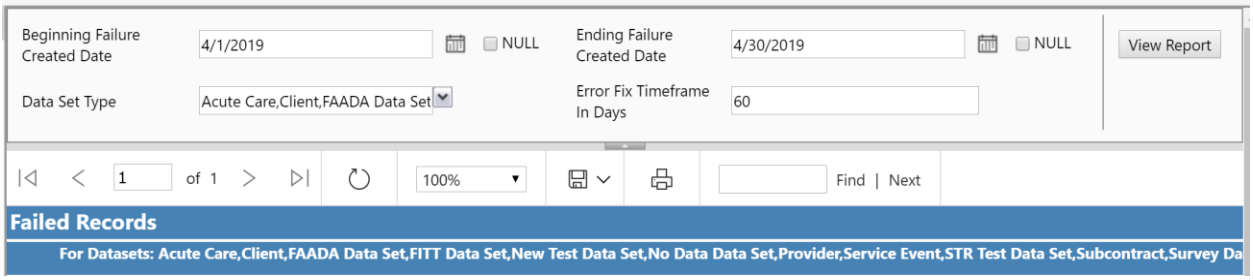


FIGURE 103 - FAILED RECORD REPORT

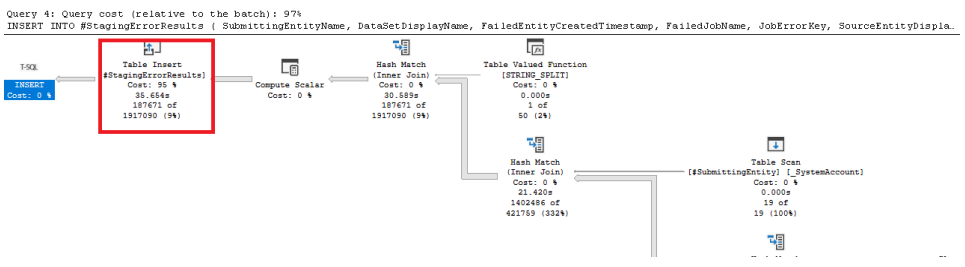
The Failed Record report provided ability to the submitting entity, DCF users, and administrators to understand the failed records and corresponding error messages for each submitting entity. The user can filter the report by the following criteria:

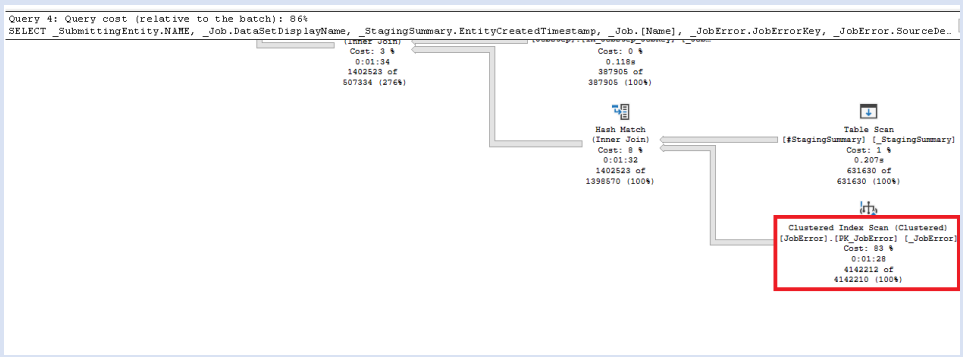
- Data Set -The user can select one or more Data Set values at a time in their filter. Selecting zero means that the report will return all failed records regardless of the data set
- Error Fix Timeframe- Defaults to 60 and indicates the number of days after the failure that the error was corrected, and the failed record successfully processed
- Beginning Failure Created Date
- Ending Failure Created Date

This report uses the results of SQL Stored Procedure "[**FindSsrsReportDatsSetForFailedRecords**]" as its dataset. Execution details of the stored procedure are given below along with parameters:

```
EXEC [SubmissionModule].[FindSsrsReportDatsSetForFailedRecords] '2019-05-01', '2019-05-31',
'FloridaDepartmentOfChildrenAndFamilies.Domain.ProviderModule.ClientDataSet',
60, 'muhammad.farhan'
```

TABLE 7 - FAILED RECORD REPORT - STORED PROCEDURE EXECUTION PLAN

Execution Details	
Estimated # of rows in result set	187,671
Time	10:13 min
Estimated # of rows to be read	631,630
High Cost Query (97%)	

Execution Details	
High Cost Index	

2. Job Submission Performance ->Submission Rollup

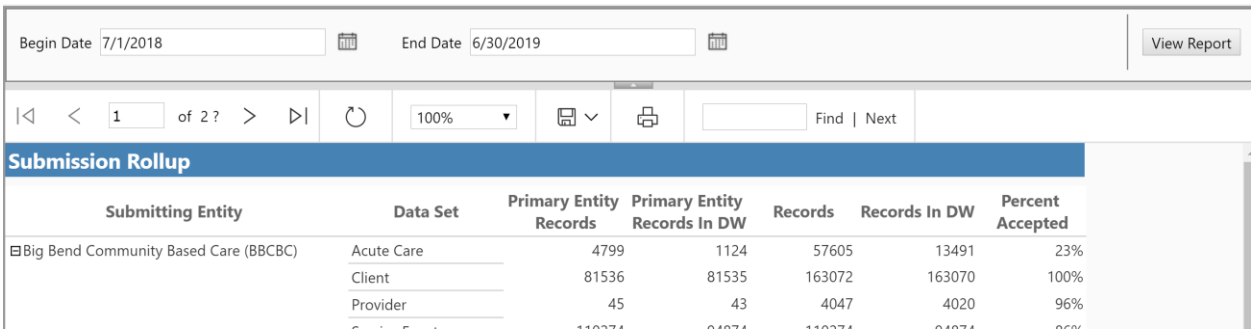


FIGURE 104 - SUBMISSION ROLLUP REPORT

This report provides information related to the submission datasets and their records submitted into DW database and the accepted percentage.

The user can filter the report by the following criteria:

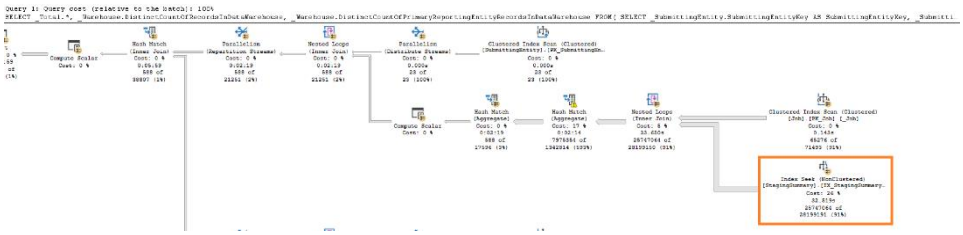
- Beginning Date
- EndDate

This report uses the results of the SQL Stored Procedure "[**FindSsrsDataSetForDetailedSubmissionRollup**]" as its dataset. Execution details of the stored procedure are given below along with parameters.

```
Exec [SubmissionModule].[FindSsrsDataSetForDetailedSubmissionRollup] '2018-07-01','2019-06-30'
```

FIGURE 105 - SUBMISSION ROLLUP REPORT - STORED PROCEDURE EXECUTION PLAN

Execution Details	
Estimated # of rows in result set	588
Time	6 min

Estimated # of rows to be read	588
High Cost Index	

3. Job Submission Performance ->Overall Job Performance

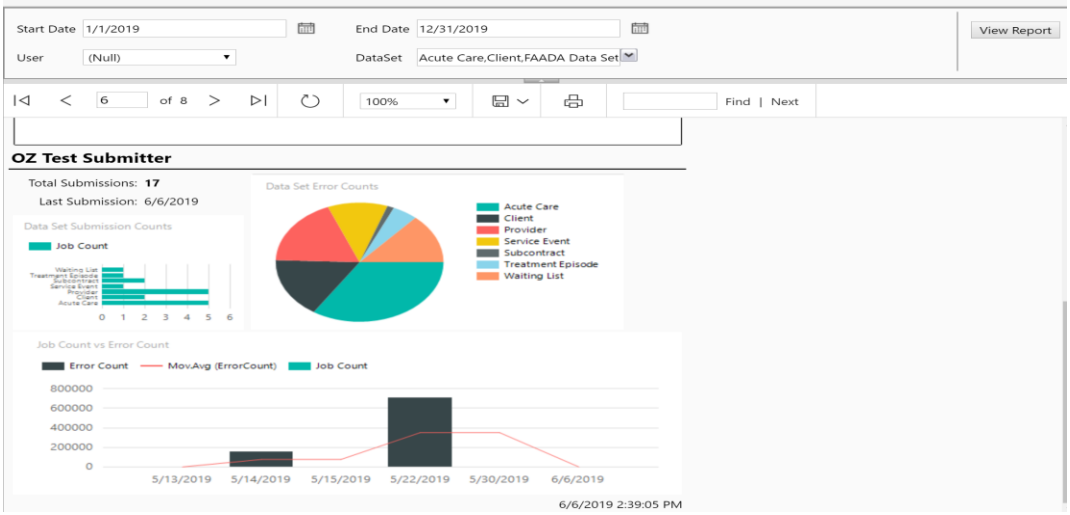


FIGURE 106 - OVERALL JOB PERFORMANCE REPORT

The Overall Job Performance Report provides administrators the ability to understand the failed records and the corresponding error messages for each submitting entity, across jobs, as well as the overall trend of their job performance. It is designed to show whether submitting entities are improving their job submissions over time.

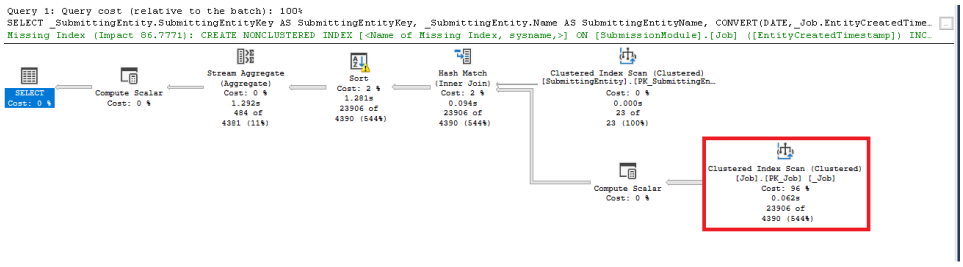
The user can filter the report by the following criteria:

- Data Set -The user can select one or more Data Set values at a time in their filter. Selecting zero means that the report will return all failed records regardless of the data set
- Beginning Job Created Date
- Ending Job Created Date

This report uses results of the SQL Stored Procedure "[**FindSsrsReportDataSetForJobOverallPerformance**]" as its dataset. Execution details of the stored procedure are given below along with parameters.

```
exec [SubmissionModule].[FindSsrsReportDataSetForJobOverallPerformance]
    '2019-01-01','2019-05-31', null
```

TABLE 8 - OVERALL JOB PERFORMANCE REPORT - STORED PROCEDURE EXECUTION PLAN

Execution Details	
Estimated # of rows in result set	484
Time	1 min
Estimated # of rows to be read	484
High Cost Query	

4. Acute Care -> DCF Purchased Bed Occupancy Rates

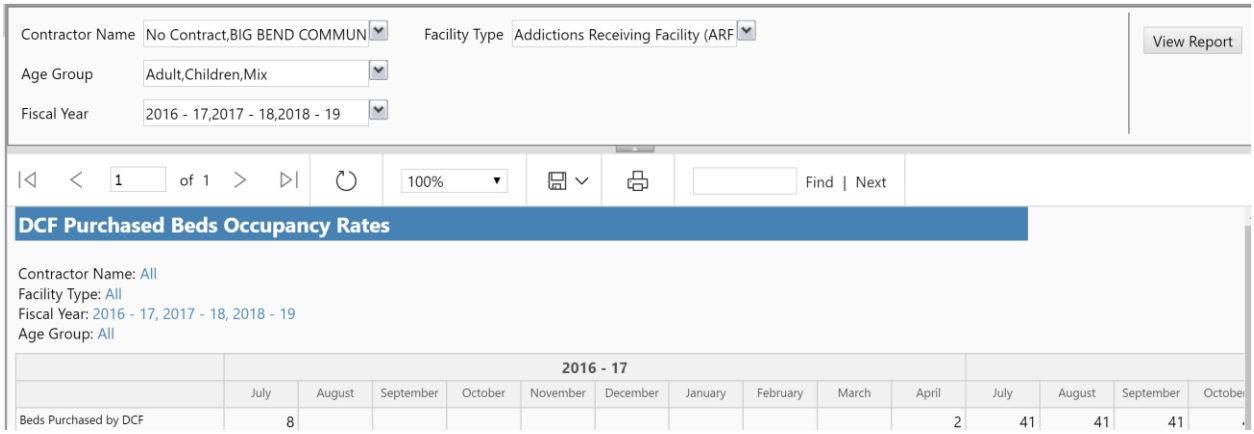


FIGURE 107 - DCF PURCHASED BEDS OCCUPANCY RATES REPORT

The purpose of this report is to provide, in tabular and graphical form, the average daily census of total beds purchased by DCF, the number of these beds occupied by DCF payor class, and the occupancy rates of these beds per month in each fiscal year.

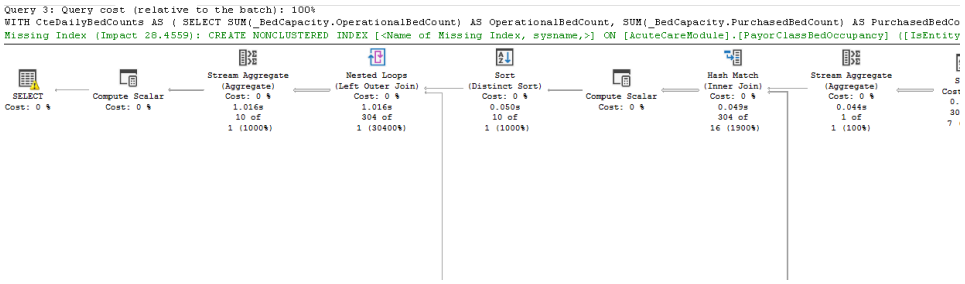
The user can filter the report by the following criteria:

- Contractor Name
- Facility Type

- Fiscal Year
- Age Group

This report uses the results of SQL Stored Procedure "[**FindSsrsReportDataSetForPurchasedBedsOccupancyRates**]" as its dataset. Execution details of the stored procedure are given below along with parameters.

```
exec [AcuteCareModule].[FindSsrsReportDataSetForPurchasedBedsOccupancyRates]
'10D286EF-F0E4-4CBA-88B0-A97500A52B77,40E0F0BA-6D7B-4D19-9545-A97500A52B77,2ACCD2B-BF38-422F-B0CC-A97500A52B6D',
'9000.3,9000.2,9000.1,9000.4,9000.5,9000.6,9000.7',
'9100.1,9100.2,9100.3','2018 - 19'
```

Execution Details	
Estimated # of rows in result set	10
Time	20 Sec
Estimated # of rows to be read	10
High Cost Query	

10.5.2. Analytics and Reporting Recommendations

Create a Separate Database for Reporting

Build a dedicated reporting database that consolidates the data from production systems (OLTP) and transforms it into a form that is much friendlier for reporting. A reporting database is a relational database that is designed for analytical and reporting rather than transactional work. Reports on operational database will use a lot of resources and will interfere with the performance of the application and reporting. A separate reporting database takes a copy of the essential operational data but stores it in a different schema.

OLTP (Online Transactional Processing) is a type of data processing that executes transaction-focused tasks. It involves inserting, deleting, or updating small quantities of database data. It is meant to:

- Handle large quantities of transactional or operational data
- Be the source system of record for the data that they're storing
- Provide data for business processes or tasks
- Respond to frequent data updates and additions
- Respond to queries quickly with relatively small record sets

A **Reporting Database** is a relational database that is designed for analytical rather than transactional work. It is meant to:

- Handle large quantities of data for reporting and analysis
- Be a consolidation point for data from one or many OLTP databases
- Provide data to help with the analysis and planning of business operations
- Provide views based on multiple dimensions that reflect business concepts
- Accept large quantities of data as fed in through repeated batch processes
- Run large and complex queries to aggregate data across multiple data dimensions

A reporting database has a number of advantages:

1. The structure of the reporting database can be specifically designed to make it easier to write reports.
2. A reporting database is not usually normalized. Data may be duplicated as necessary to make queries and reporting easier.
3. The development team can refactor the operational database with minimal impact to the reporting database.
4. Queries run against the reporting database don't add to the load on the operational database.
5. You can store derived data in the database, making it easier to write reports that use the derived data without having to introduce a separate set of derivation logic.
6. Since the reporting database is primarily for searches, indexes can be customized for better performance. (The live database would have inserts which would be affected adversely by certain indexes)

10.5.3. SQL Server Analysis Services (SSAS)

SSAS Tabular models in Analysis Services are databases that run in-memory or in DirectQuery mode, connecting to data directly from back-end relational data sources. By using state-of-the-art compression algorithms and multi-threaded query processor, the Analysis Services Analytics engine delivers fast access to tabular model objects and data by reporting client applications like Power BI, SSRS and Excel.

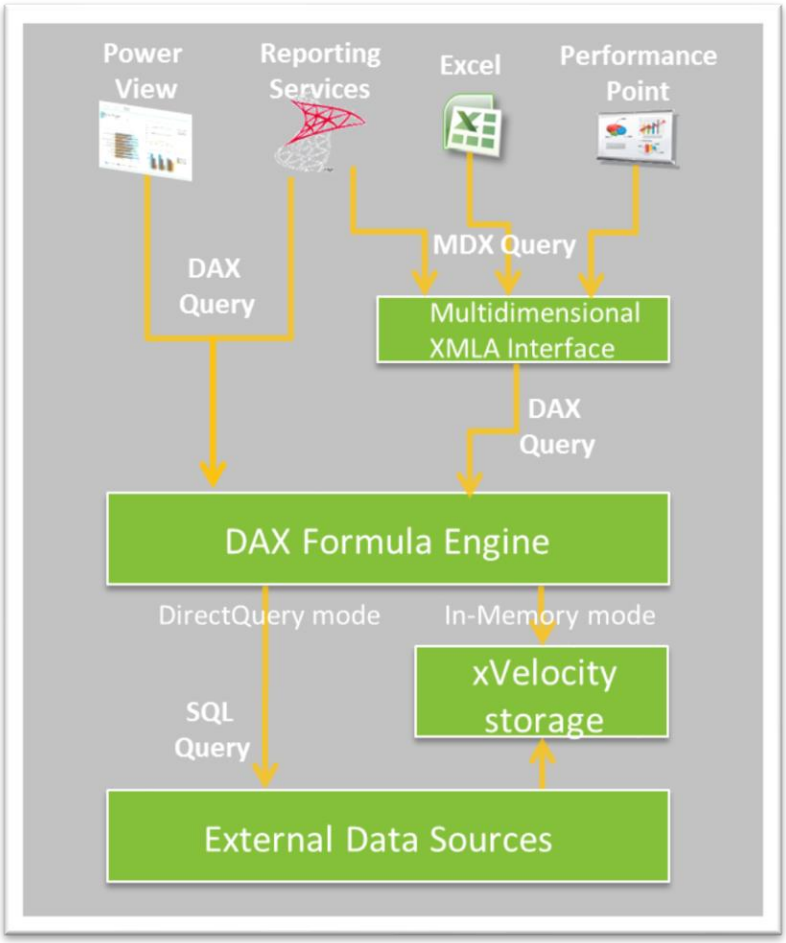


FIGURE 108 - SSAS MODEL

FASAMS Analysis Services Solution

The OZ Team was given source code for the two SSAS project "Bird.SSAS.Model" and "Fasams.DataTier". The OZ team reviewed the different components of the both SSAS solutions.

Ad-hoc report development functionality is available for DCF users with the appropriate permissions. FASAMS published a Microsoft SQL Server Analysis Services (SSAS) tabular reporting model, which exposed the entities in the FASAMS ODS database. These features are supported by Microsoft SQL Server Analysis Services (SSAS) tabular model and Report Builder. Furthermore, some FASAMS out-of-the-box SSRS reports are also using SSAS tabular model as a source data set, like "Outcome Measures Report - Cumulative Data Set Size", "Outcome Measures Report - Cumulative Performance", and "Outcome Measures Report - Monthly Data Set Size".

FASAMS SSAS tabular database runs in-memory, connecting to data directly from back-end relational data sources. All the data is loaded in memory and all queries/reports are answered from there.

The findings of SSAS data model are covered in the following sections.

FASAMS SSAS DATA Model

There is no separate SSAS data mart design for FASAMS application, the transactional database (OLTP) entities (table, views, and SPs) are consumed in SSAS model. FASAMS SSAS model schema is a combination of snowflake and star. Many dimension tables are normalized which splits data into additional tables.

Star Schema

The star schema is the simplest type of Data Warehouse schema. It is known as star schema as its structure resembles a star. In the Star schema, the center of the star can have one or more fact tables having any number of associated dimension tables. It is also known as Star Join Schema and is optimized for querying large data sets.

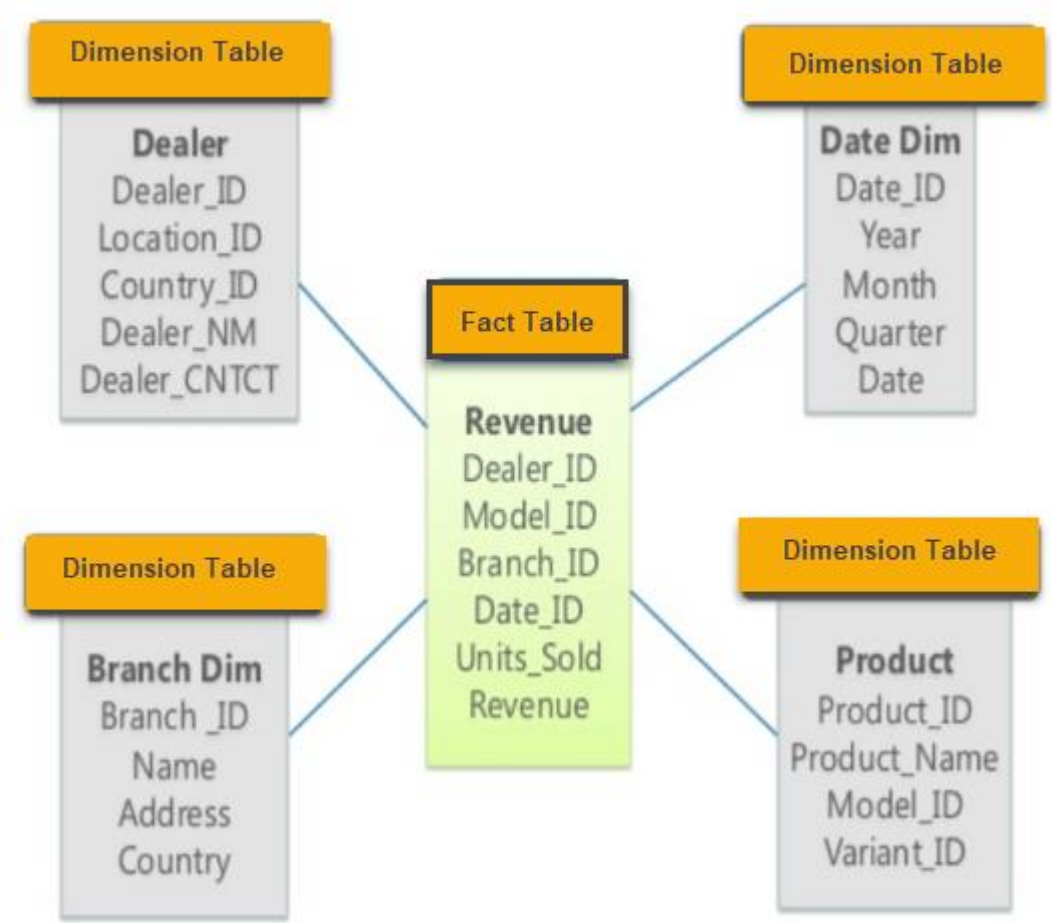


FIGURE 109 - STAR SCHEMA SAMPLE

6.5.3.2.2 Snowflake Schema

A Snowflake Schema is an extension of a Star Schema, and it adds additional dimensions. It is called snowflake because its diagram resembles a Snowflake. The dimension tables are normalized which splits data into additional tables. In the following sample, Country is further normalized into an individual table.

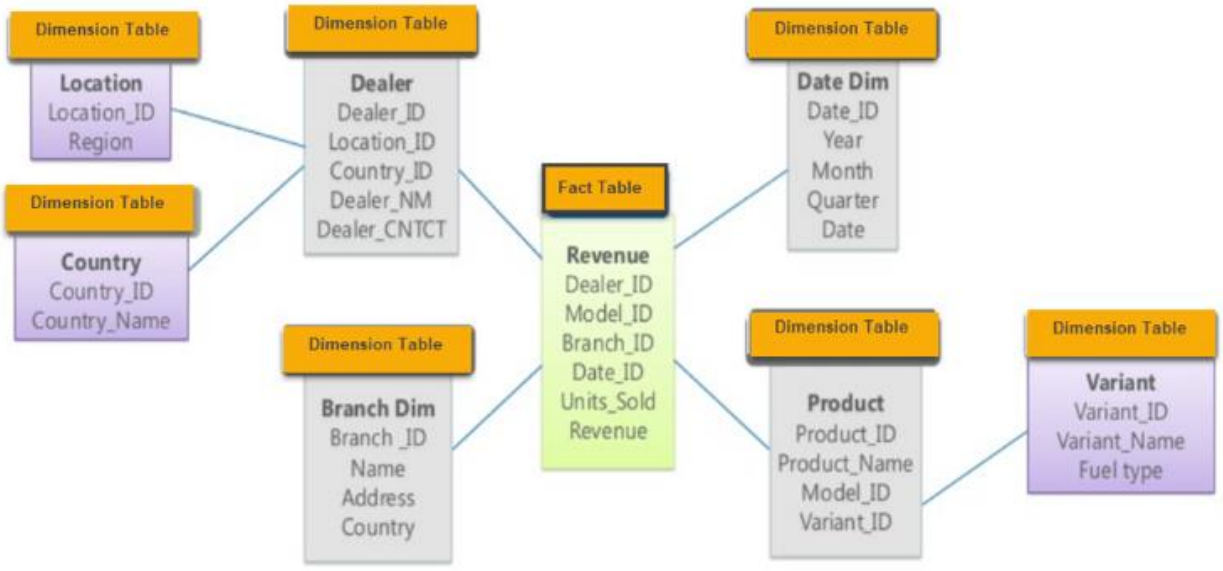


FIGURE 110 - SNOWFLAKE SCHEMA SAMPLE

FASAMS SSAS DATA Model Schema

The FASAMS SSAS model is a snowflake schema. Below are some examples where dimensions are further normalized into multiple tables.

Note: The best practice is to denormalize these multiple tables into a single flat dimension table.

1. Bed Dimension
   Bed dimension is normalized into 4 tables.

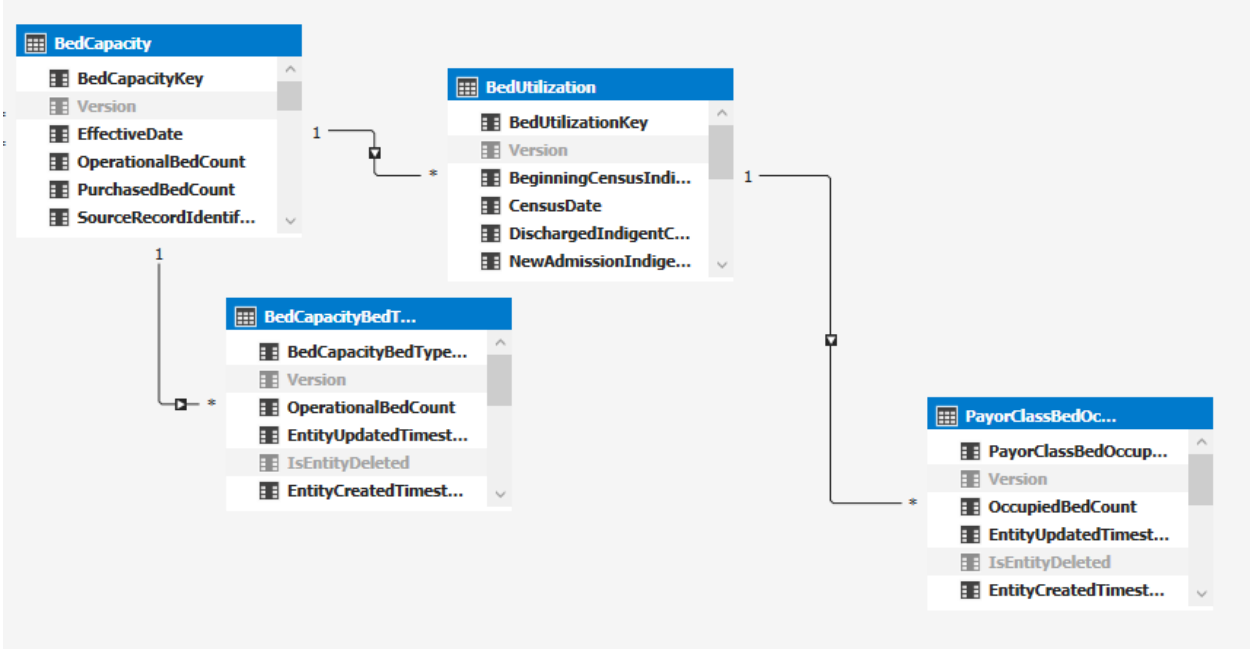


FIGURE 111 - BED DIMENSION

2. Provider Site Dimension

Provider site dimension is normalized into 4 tables.

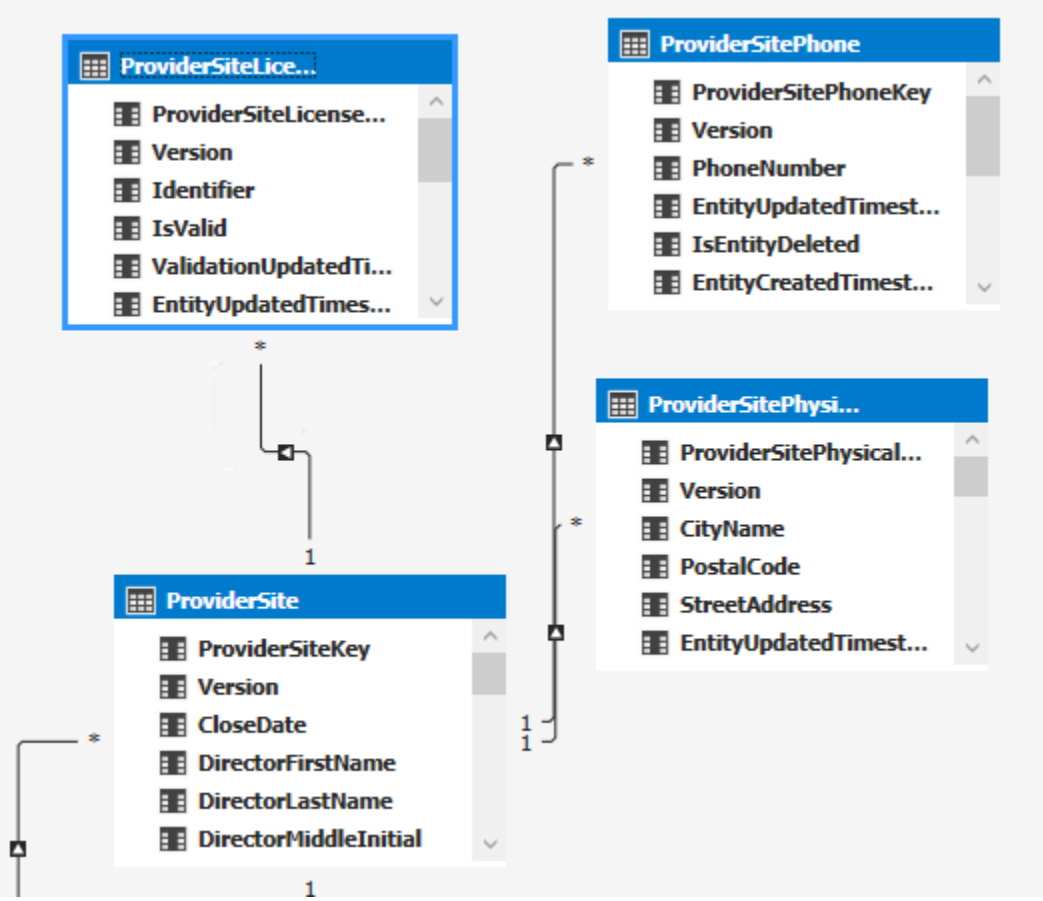


FIGURE 112 - PROVIDER SITE DIMENSION

3. Contract Dimension

Contract dimension is normalized into 2 tables.

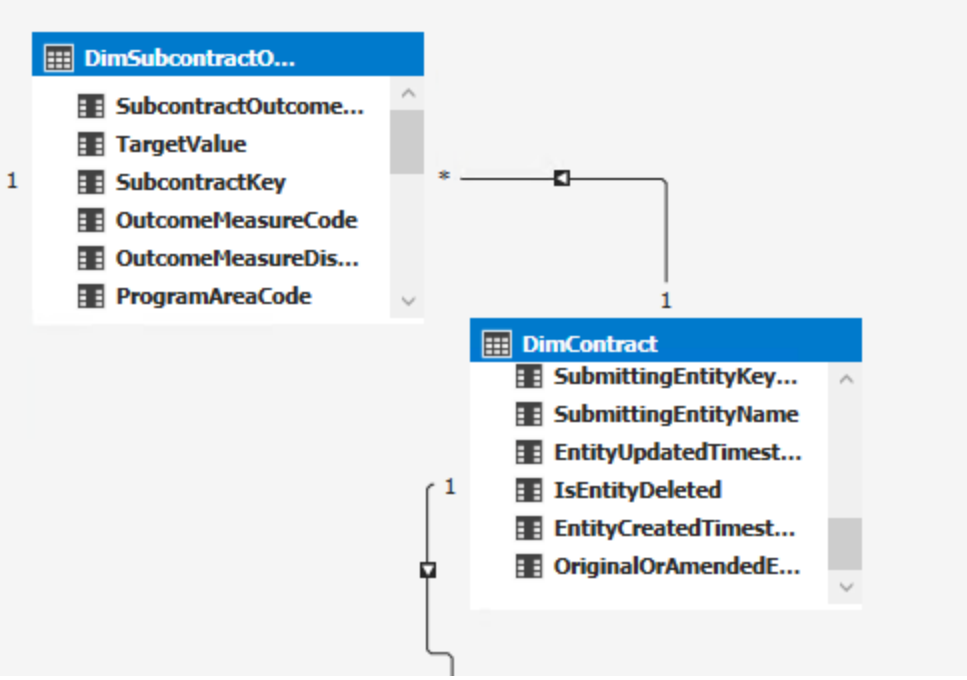


FIGURE 113 - CONTRACT DIMENSION

SSAS Data Model Recommendations

Convert Snowflake into Star Schema

SSAS Tabular works great in small data models, but as the size of your models increase, so does the number of data performance issues. Theoretically, the best way to extract great performance from SSAS Tabular is to flatten the entire data set into one single table. This could lead to a lengthy and expensive ETL process and is not advisable. Thus, the best practice here is to have a **Star Schema** with the data fields merged into either the fact table or dimension tables, such that the dimensions carry similar or related information.

Start Schema Sample: ALL the DIMENSION TABLES are DIRECTLY related to FACT TABLE.

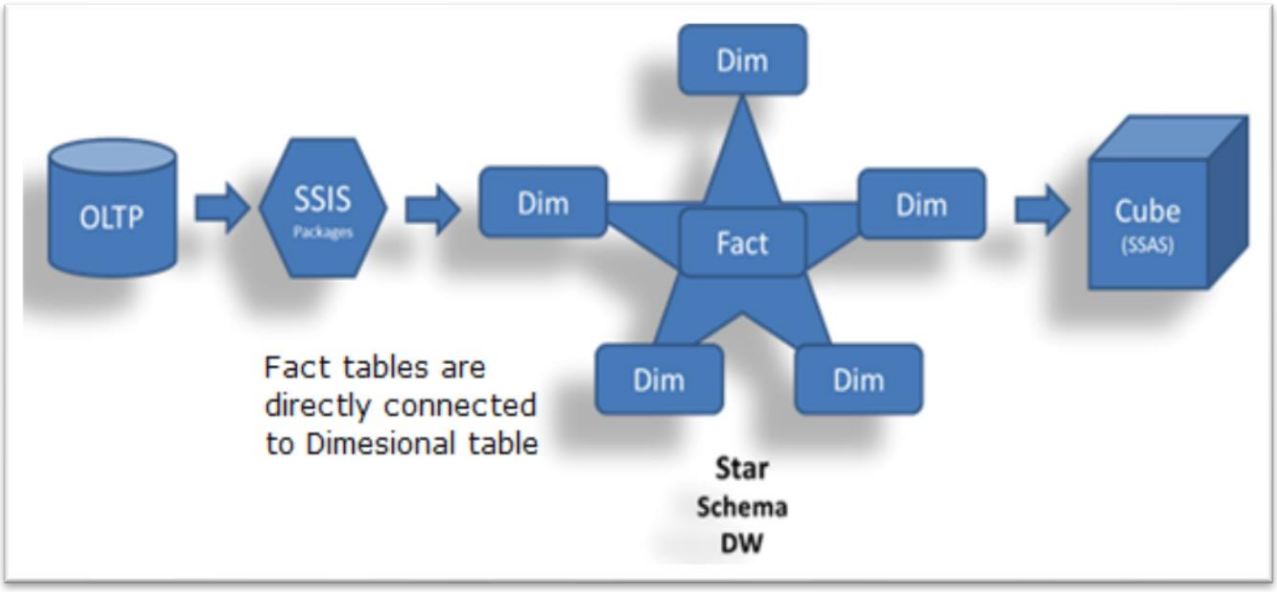


FIGURE 114 - STAR SCHEMA SAMPLE

TABLE 9 - STAR AND SNOWFLAKE SCHEMA COMPARISONS

Star Schema	Snowflake Schema
Centrally located fact table surrounded by denormalized dimension table	Centrally located fact table surrounded by the normalized dimension table.
In Star Schema, all dimensions will be linked directly with the fact table	In Snowflake Schema, some dimensions are linked directly to the fact table and some dimensions are indirectly linked to fact tables (with the help of middle dimensions)
Simple DB design	Complex DB design
Data aggregated in single dimension table	Data split into different dimension tables
High level of data redundancy	Very low data redundancy
Simple query execution at cube processing	Complex joins performed at Cube Processing
Query results faster (increases the query performance because we can extract the data with fewer number of joins)	Delay in query processing due to complex joins

For optimal performance, it's recommended to choose Star Schema over Snow-flake Schema.

Minimize the number of columns

SSAS Tabular uses its column-store technology to store data. This implies all data is stored by column and pulling an entire table can remarkably decrease the efficiency of the model depending on the table size. Thus, it is advised to only pull in selected columns that are the most relevant and are needed in the model. It is advisable to never import an entire table. Even if the columns are hidden in the view, the data still occupies memory, thereby, slowing down the performance.

The lower the cardinality, the better the performance

Cardinality or granularity refers to the unique values in a column. Having a higher granularity leads to a larger column dictionary, increasing the in-memory usage. This, in turn, decreases the compression efficiency, thereby affecting performance. Increasing the rows will not impact the performance as much, as long as the cardinality is not too high.

Use less expensive data types

Certain datatypes are more expensive for SSAS Tabular to process than others. For example, using Integer IDs in place of String and Date instead of Date Time can significantly improve the performance of our data model. Storing time only down to the granularity that is really needed is better, for instance, if milliseconds are not required, don't store them.

Hardware Considerations

As SSAS Tabular is a memory intensive application, it is important to consider various factors when hardware selections are made. A few recommendations are listed below:

- **CPU Speed**: Faster CPU helps in computing results faster.
- **CPU Cores**: CPU cores help in managing concurrent user loads. However, having more CPU cores can lead to a decrease in CPU speed. Thus, a balance between the CPU speed and CPU cores must be established.
- **CPU Cache**: The more, the better as retrieving data from the CPU cache is much faster than retrieving data from RAM.
- **RAM**: A RAM of at least 2.5 times the size of the model is recommended for optimum performance if the model is to be processed on the same server.
- **RAM Speed**: For a memory-intensive application like SSAS Tabular, the faster the RAM speed, the better the performance.

10.6. Security, Hardware, Network & Software

FASAMS security is internally being managed by the DCF Network and security team. FASAMS is only accessible via the Virtual Private Network outside the DCF domain. However, to assess the security architecture following few points would be considered:

- What are the data entity and attribute access rules which protect the data from unintentional and unauthorized alterations, disclosure, and distribution?
- What are the data protection mechanisms to protect data from unauthorized external access?
- What are the data protection mechanisms to control access to data from external sources that temporarily have internal residence within the enterprise?

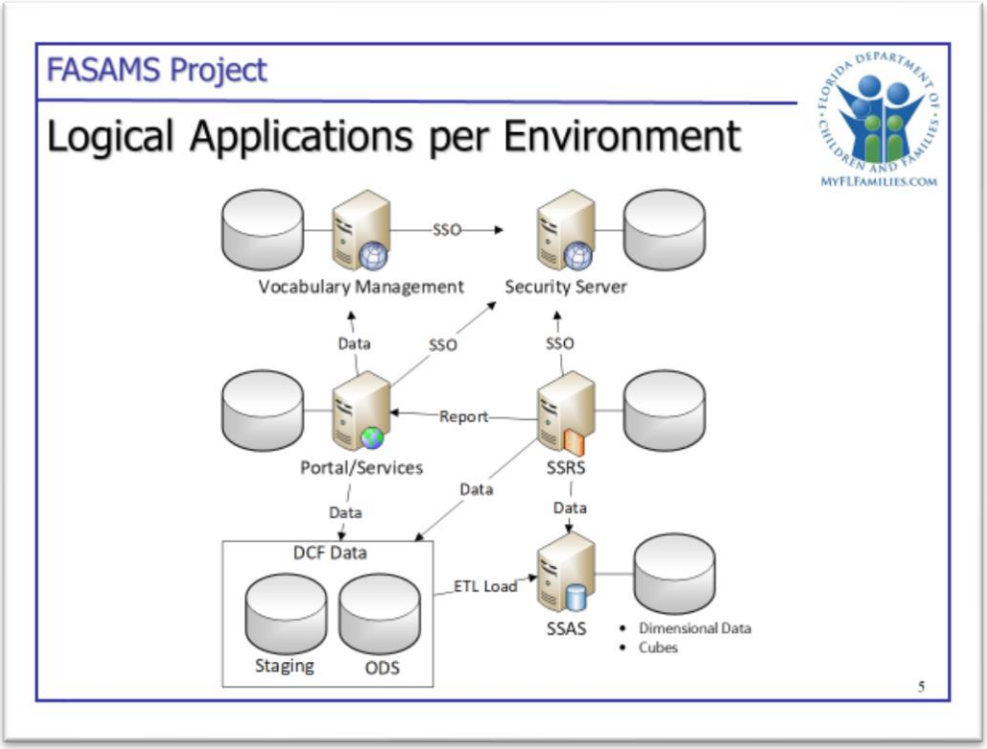


FIGURE 115 - LOGICAL APPLICATIONS PER ENVIRONMENT

- **Security Awareness:** Have you ensured that the corporate security policies and guidelines to which you are designing are the latest versions? Have you read them? Are you aware of all relevant computing security compliance and risk acceptance processes?
- **Identification/Authentication:** Diagram the process flow of how a user is identified to the application and how the application authenticates that the user is who they claim to be. Provide supporting documentation to the diagram explaining the flow from the user interface to the application/database server(s) and back to the user. Are you compliant with corporate policies on accounts, passwords, etc.?
- **Authorization**: Provide a process flow from beginning to end showing how a user requests access to the application, indicating the associated security controls and separation of duties. This should include how the request is approved by the appropriate data owner, how the user is placed into the appropriate access-level classification profile, how the user ID, password, and access is created and provided to the user. Also include how the user is informed of their responsibilities associated with using the application, given a copy of the access agreement, how to change password, who to call for help, etc.
- **Access Controls:** Document how the user IDs, passwords, and access profiles are added, changed, removed, and documented. The documentation should include who is responsible for these processes.
- **Sensitive Information Protection**: Provide documentation that identifies sensitive data requiring additional protection. Identify the data owners responsible for this data and the process to be used to protect storage, transmission, printing, and distribution of this data. Include how the password file/field is protected. How will users be prevented from viewing someone else's sensitive information? Are there agreements with outside parties (partners, suppliers, contractors, etc.) concerning the safeguarding of information? If so, what are the obligations?
- **Audit Trails and Audit Logs**: Identify and document group accounts required by the users or application support, including operating system group accounts. Identify and document individual accounts and/or roles that have superuser type privileges, what these privileges are, who has access to these accounts, how access to these accounts is controlled, tracked, and logged, and how password change and distribution are handled, including operating system accounts. Also identify audit logs, who can read the audit logs, who

can modify the audit logs, who can delete the audit logs, and how the audit logs are protected and stored. Is the user ID obscured in the audit trails?

- **External Access Considerations**: Will the application be used only internally? If not, are you compliant with corporate external access requirements?

In this section of the document we will review the FASAMS architecture in detail for the provided components where we have access and will list the OZ Technical Team findings for different tiers of the project. This section will address Network, Hardware, Software, Security, Reports and Data Warehouse. Each section will present the current architecture diagram extracted from the FEI or DCF provided documentation - we will add our findings on the code for the related tier which is provided.

Hardware, Network and Software requirements and design is the key in application performance and maintainability. The OZ Technical team wanted to assess these areas of the FASAMS application hence the access to AWS Cloud environment and the FASAMS Web Application source code not provided which restricts the review. However, in this section described key areas which would be considered while performing architecture level review. Based on the walkthrough session with FEI Systems on Amazon environment, it was communicated that the FASAMS application is hosted in AWS Virtual Machine which can only be accessed using DCF Virtual Private Network.

Below are a few key areas which need to be completed to have a comprehensive application review:

- What are the hardware requirements (machines, CPU, RAM, Storage, etc.)?
- Which environments are required (for example: Development, Testing, Staging, etc.)?
- Does the architecture need to be deployed in cloud (private or public)? Is there a legal requirement to host and process data in certain Amazon regions?
- What are the OS requirements?
- What are the 3rd party software requirements?
- Do you need agents to monitor the machine/application?
- Does it require balancing?
- Does it require session persistence?
- Do we have enough network capacity (ports, bandwidth, etc.) for all network devices (switches, routers, etc.)?
- What key issues have been identified or analyzed that the project believes will drive evaluations of hardware and operating systems for networks, servers, and end-user devices?
- Which system capabilities will involve high-volume and/or high-frequency data transfers?
- How does the system design impact or involve end-user devices?
- What is the quantity and distribution of usage, data storage, and processing?
- Describe the process or tools available for checking that the system is properly installed.
- Describe tools or instrumentation that are available for monitoring the health and performance of the system.
- Describe the tools or process in place that can be used to determine where the system has been installed.
- Describe what form of audit logs are in place to capture system history, particularly after a mishap.
- Describe the capabilities of the system to dispatch its own error messages to service personnel.
- Is the flexibility of the architecture demonstrated?
- How can it cope with likely changes in the requirements?
- What is the user data backup frequency and expected restore time?

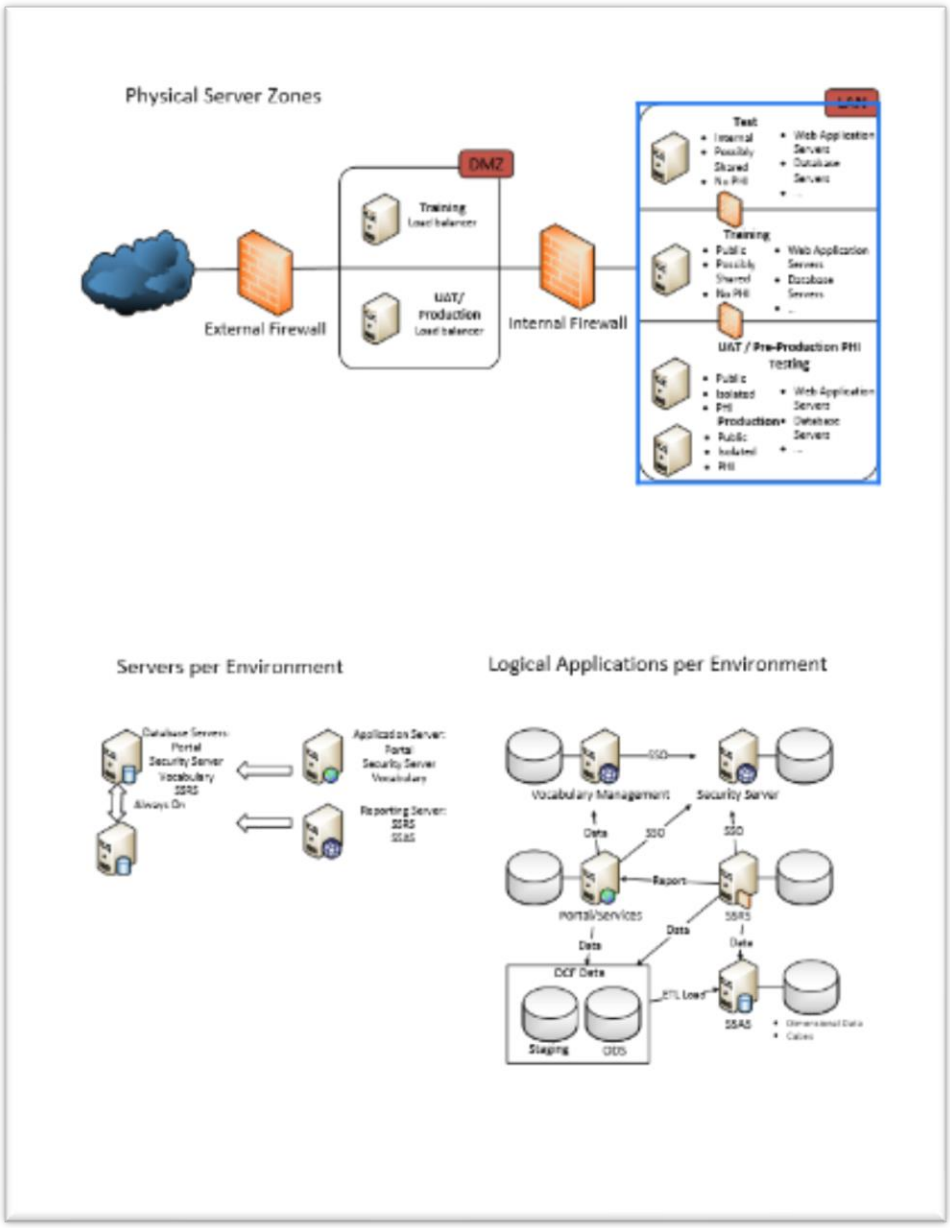


FIGURE 116 - PHYSICAL SERVER ZONES

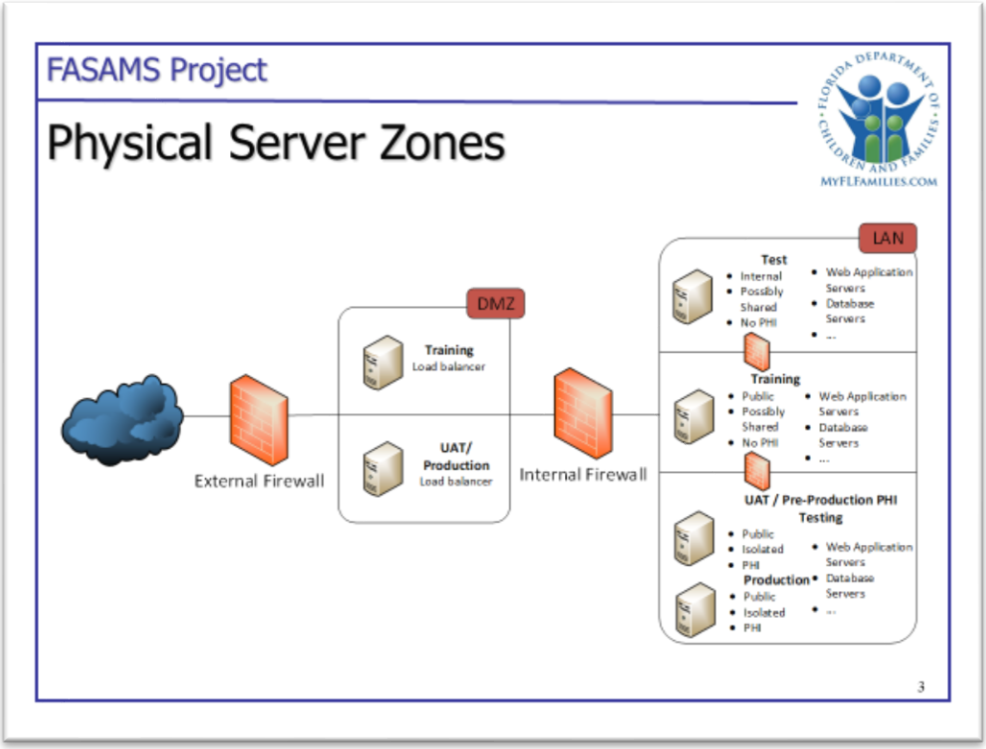


FIGURE 117 - PHYSICAL SERVER ZONES

As the OZ Team has not been provided access to FASAMS (Web Application/Entry Point/source code), it is difficult to assess the software different tiers and layers however following are the few questions which need to be addressed in order to review the architecture.

Since FASAMS is being accessed by using DCF VPN, VPNs introduce an additional leg to the path taken by data packets and can sometimes have a negative impact on network latency. The increased latency can possibly affect application performance. The following scenarios can make an impact on application performance:

- Increased Network Latency
- Application Unavailability
- Loss of Internet Connection

If the OZ team had appropriate access, we would perform the actions in the Software Services and Middleware Checklist below:

- How error conditions are defined, raised, and propagated between application components?
- The general pattern of how methods are defined and arranged in various application modules.
- The general pattern for how method parameters is defined and organized in various application modules. Are [in], [in/out], [out] parameters always specified in the same order? Do Boolean values returned by modules have a consistent outcome?
- The approach that is used to minimize the number of round-trips between client and server calls, particularly for out-of-process calls, and when complex data structures are involved.
- Major data structures that are passed between major system components.
- Major communication protocols that are used between major system components.
- To what extent the system is designed with stateful and stateless components.
- How and when state is saved for both stateful and stateless components.
- The extent to which objects are created, used, and destroyed versus re-used through object pooling.

- The extent to which the system relies on threading or critical section coding.
- The approach and the internal documentation that is used internally in the system to document the methods, method arguments, and method functionality.
- Code review process that was used to build the system.
- Unit testing that has been used to test the system components.
- Pre and post-condition testing that is included in various system modules.
- Assertion testing that is included with the system.
- Do components support all the interface types they need to support or are certain assumptions made about what types of components will call other components either in terms of language bindings or other forms of marshaling?
- Whether the software needs to check for floating-point round-off errors.
- What tools or processes have been used to test the system for memory leaks, reachability, or general robustness.

Below is the Software Architecture diagram extracted from one of the FEI documents libraries, this document is describing at high level of application layers and segregation of tiers i.e. for Application, Web, and Database Server.

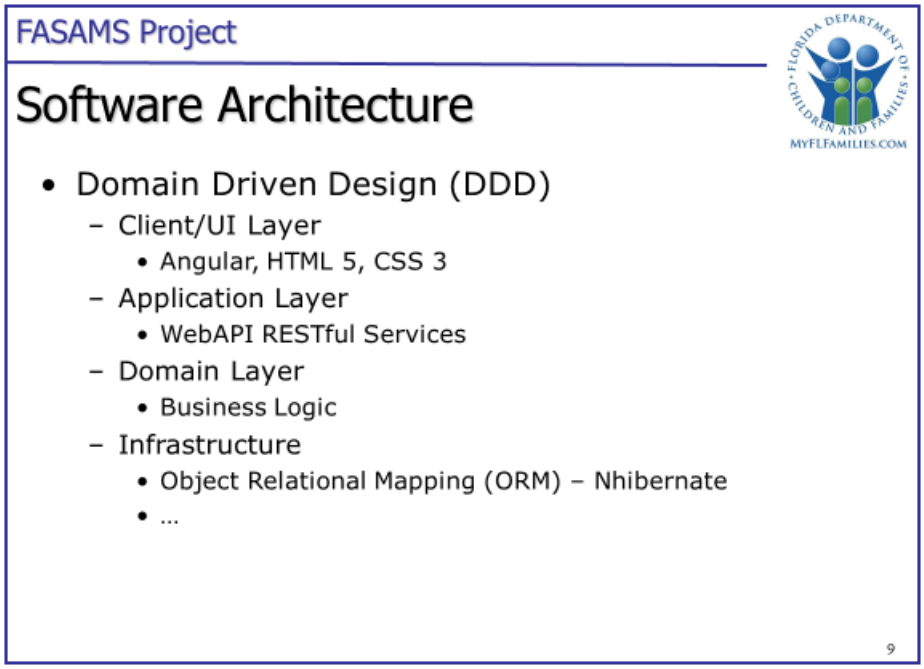


FIGURE 118 - SOFTWARE ARCHITECTURE

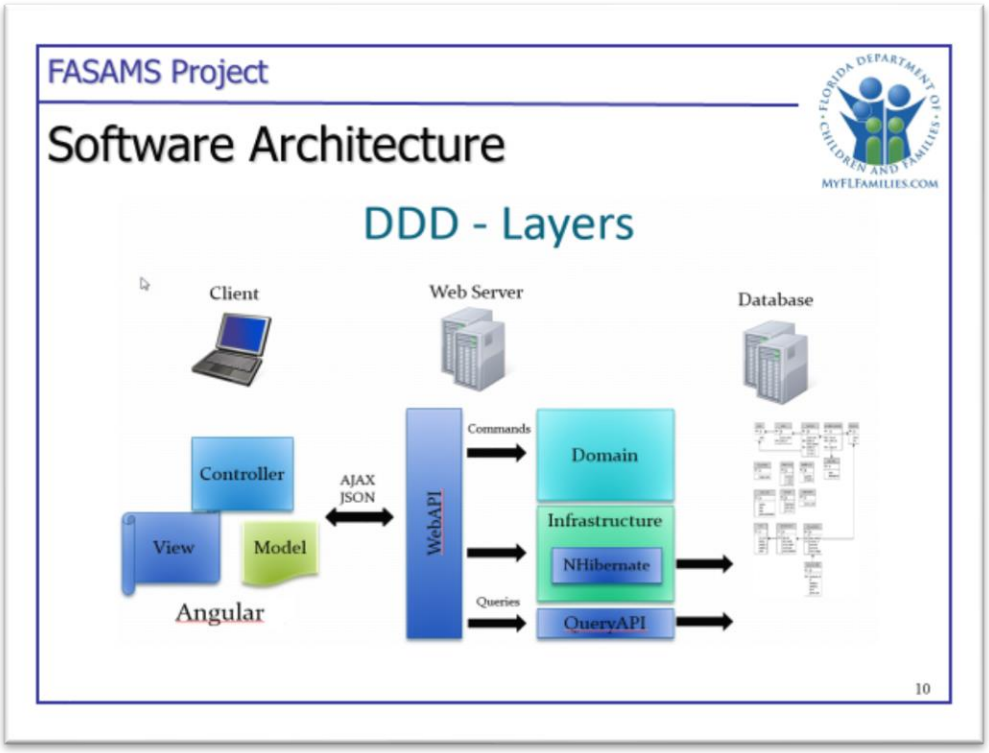


FIGURE 119 - SOFTWARE ARCHITECTURE

- To what extent the system components are either loosely coupled or tightly coupled?
- How the system and its components are designed for refactoring.

Negative indicators:

- **High complexity**: a component has a complex interface or functionality.
- **Low cohesion**: a component contains unrelated functionality.
- **High coupling**: two components have many (mutual) connections.
- **High fan-in**: a component is needed by many other components.
- **High fan-out**: a component depends on many other components.

10.7. Recommendations/Next Steps

10.7.1. Database

Logging

The log files contain more than 10 million rows, which can severely impact performance. This is likely the most critical issue that needs to be fixed. Properly configure the logging framework to:

- Automatically roll over and purge log entries
- If appropriate, save log entries to text files instead of the database
- Log entries do not need to be stored in fault-tolerant databases
- Most log entries are used as information. Either disable the code lines that are creating these entries or modify it to make the entries more usable.
- It looks like the production instance is running at the trace level of logging; reduce the logging severity level to warning or debug.

Create Table Partitions for large tables

Partitioning is the database process where very large tables are divided into multiple smaller parts. By splitting a large table into smaller, individual tables, queries that access only a fraction of the data can run faster because there is less data to scan. The main goal of partitioning is to aid in maintenance of large tables and to reduce the overall response time to read and load data for particular SQL operations (click here for detail).

Optimize Database Indexes

Indexing is generally the easiest way to improve the performance of your SQL Server queries. However, if not designed right, it might hurt your application performance negatively as well. For instance, if you have an OLTP system, then you should review all the newly created indexes carefully to determine if they are really needed. Limiting the number of indexes is important as well, else it will slow down the Data Manipulation operations like INSERT/DELETE/UPDATE. Not to forget the additional disk space the indexes consume (click here for detail).

10.7.2. Create a separate database for analytics and reporting

Generate a new database for reporting purposes. All reports need to get data from this database as it is designed specifically for this purpose. A nightly job can be scheduled to transfer OLTP data using SSIS to the reporting database. Data will be organized and calculated during data migration to make it readily available for reporting. This means that when a report is generated, retrieval of data from the database will be fast as it will involve minimum joins and no calculations (click here for detail).

10.7.3. Implement Power BI For Ad hoc Reporting

Microsoft's Power BI provides ad-hoc computing functionality. SQL Server Analysis Services Tabular model, Azure Analysis Services, and Power BI share the same underlining fundamentals and principles, so Power BI has great compatibility with SSAS tabular Model.

Power BI benefits from the connectivity of Microsoft platforms — reports can be embedded and exported into a number of external applications. Reports published via SharePoint host built-in notifications that alert users to any fluctuations in the data. Microsoft offers community forums and other resources to help educate users on ad hoc reporting with this top-branded platform.

Power BI is a collection of software services, apps, and connectors that work together to turn your unrelated sources of data into coherent, visually immersive, and interactive insights. Your data may be an Excel worksheet, or a collection of cloud-based and on-premises hybrid data warehouses. Power BI lets you easily connect to your data sources, visualize and discover what's important, and share that with anyone or everyone.

FIGURE 120 - MS POWER BI

Power BI consists of:

- A Windows desktop application called **Power BI Desktop**
- An online SaaS (*Software as a Service*) called the **Power BI service**
- Power BI **mobile apps** for Windows, iOS, and Android devices

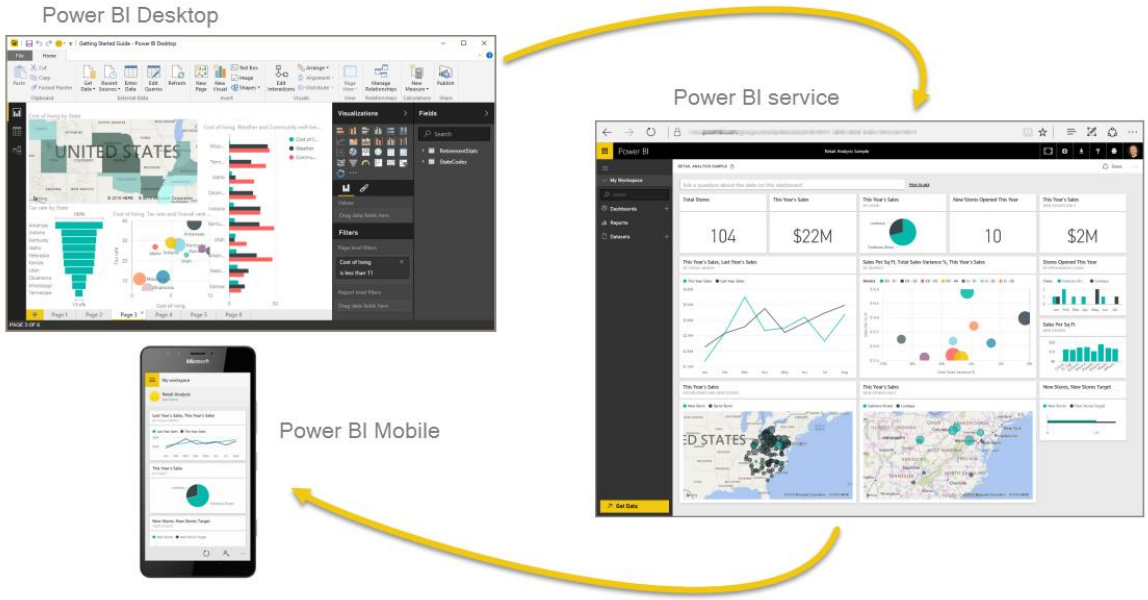


FIGURE 121 - POWER BI PARTS

These three elements—Power BI Desktop, the service, and the mobile apps—are designed to let people create, share, and consume business insights in the way that serves them, or their role, most effectively.

A fourth element, Power BI Report Server, allows you to publish Power BI reports to an on-premises report server, after creating them in Power BI Desktop. Read more about Power BI Report Server.

10.7.4. Separate Staging Database

The current database includes staging objects whereas these should not be part of the transactional database. Staging objects should have their own separate database.

11. Amendment 1 - Source Code Review

The OZ team was given FASAMS application source code to perform code review activity. Initially the provided source code was not compilable because some dependency DLLs and References were missing. OZ team requested DCF to provide source code for missing references in order to start the code review process. Later, missing DLL files were provided to the OZ team to use in the solutions and to perform FASAMS static code review.

This section is the amendment of the original report which was submitted on Friday June 28, 2019.

11.1. Summary

In continuity of the FASAMS User Experience and Technical Review, the OZ team was requested to perform the FASAMS application source code review. Initially source code was provided by DCF for the SSRS Reports (RDL) and

SSAS (Model) where OZ team reviewed and added analysis in the FASAMS Technical Review section. However, DCF later provided the Source Code for other components which are the entry point of FASAMS application and requested OZ team to perform the code review for these components as well.

FASAMS solution is highly dependent on external libraries as explained in section Pillar DLLs. Source code for these binaries has not been provided to the OZ team however the DLLs are shared to use in the solution. A source code change in the future can require Pillar binaries to be updated or to support the intended change with the current available DLL version.

The FASAMS architecture is a domain driven design; Domain-driven design (DDD) is an approach to software development for complex needs by connecting the implementation to an evolving model.

- User Interface layer is built using Angular, HTML and CSS.
- Application layer includes Restful services.
- Business logic is written in Domain layer.
- Infrastructure layer is using NHibernate.

SSAS and SSRS security extensions are constructed to override the default security in order to allow more control over the security and to use the custom security features utilizing the FASAMS security database and the infrastructure.

A separate solution for the data tier is created in order to have all the database(s) related artifacts and to manage, control and deploy on different environments like Dev, UAT and Production.

Windows services are considered to perform several jobs like Files processing over SFTP, Submitted files processing and import/export data for external systems i.e. FSFN. Configurations are stored in configuration files to allow these windows services to perform tasks at pre-configured schedules.

Overall, our assessment is that the code is well structured and organized and had symmetricity across all the projects in different solutions. Most projects also have their unit test projects in the source.

There are a few potential code quality issues which do not seriously impact the functionality but should have been fixed. In the current state, these issues represent technical debt in our opinion. There are also instances where the language or tool features could have been utilized, but custom code has been written instead. There are few hard-coded configurations/values i.e. SSAS server URL pointing towards UAT box which means in order to host in prod, these hard-coded configuration values have to be updated in the code and a new build needs to be created. The best practice is to always deploy the build that was tested in UAT.

It was communicated to the OZ team that issues are still being reported for this system, and for a number of these errors, a root cause analysis is still pending. We recommend that a root cause analysis for all such issues be performed using one or more of the following techniques:

- 5 Whys
- Analyze application logs
- Data analysis
- Debug application code
- Build verbose logging in modules that don’t have correct verbosity level

We also recommend a detailed analysis, planning and correction of the Amazon AWS infrastructure. The current tiering possibly falls short on several counts.

11.2. FASAMS Solutions

11.2.1. Pillar.Ssas.OpenIdConnectSecurityProxy

This is an Http Handler for custom authentication. The handler is a proxy for the SSAS data pump. It implements HTTP Basic Authentication with a custom identity store, and rewrites the XMLA BeginSession request, putting the userName from the credentials into the CustomData field, so it can be used in custom cube security. This HTTP Handler performs its own security, and the web site it's used in should be configured with only Anonymous logins. The handler will send the HTTP 401 response (unauthorized) on unauthenticated requests and will process basic credentials and store the user's identity in ASP.NET session state.

Microsoft SQL Server Analysis Services [5] supports a thin-client architecture. The Analysis Services calculation engine is entirely server-based, so all queries are resolved on the server. As a result, only a single round trip between the client and the server is required for each query, resulting in scalable performance as queries increase in complexity.

The native protocol for Analysis Services is XML for Analysis (XML/A). Analysis Services provides several data access interfaces for client applications, but all of these components communicate with an instance of Analysis Services using XML for Analysis.

Several different providers are provided with Analysis Services to support different programming languages. A provider communicates with an Analysis Services server by sending and receiving XML for Analysis in SOAP packets over TCP/IP or over HTTP through Internet Information Services (IIS). An HTTP connection uses a COM object instantiated by IIS, called a data pump, which acts as a conduit for Analysis Services data. The data pump does not examine the underlying data contained in the HTTP stream in any way, nor are any of the underlying data structures available to any of the code in the data library itself.

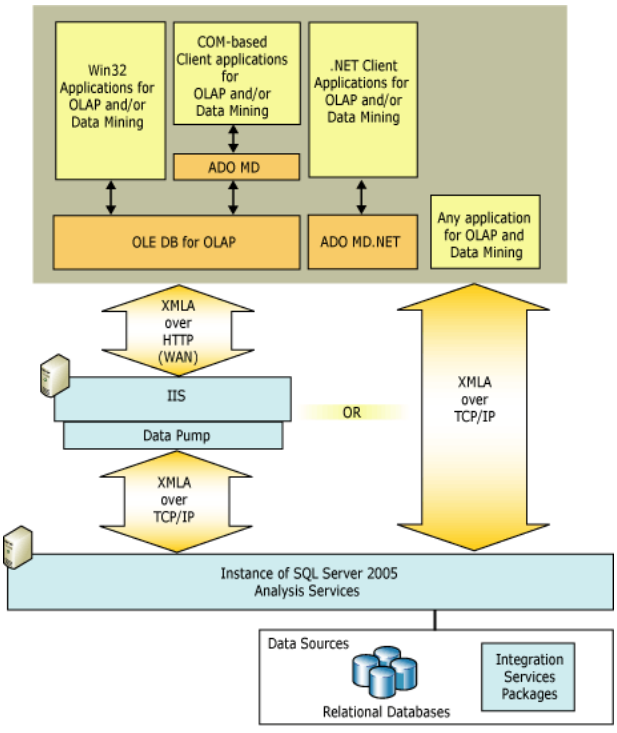


FIGURE 122 – ANALYSIS SERVICE – CLIENT ARCHITECTURE EXAMPLE

Win32 client applications can connect to an Analysis Services server using OLE DB for OLAP interfaces or the Microsoft® ActiveX® Data Objects (ADO) object model for Component Object Model (COM) automation languages,

such as Microsoft Visual Basic®. Applications coded with .NET languages can connect to an Analysis Services server using ADOMD.NET.

Analysis Services has a Web architecture with a fully scalable middle tier for deployment by both small and large organizations. Analysis Services provides broad middle tier support for Web services. ASP applications are supported by OLE DB for OLAP and ADO MD, ASP.NET applications are supported by ADOMD.NET. The middle tier, illustrated in the Figure 123 - Analysis Service - Web Architecture Example, is scalable to many concurrent users.

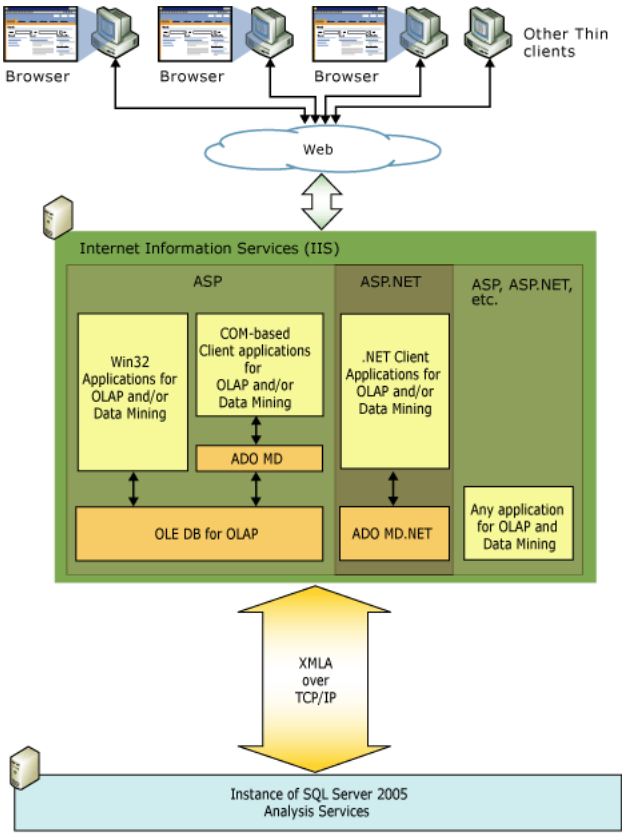


FIGURE 123 - ANALYSIS SERVICE - WEB ARCHITECTURE EXAMPLE

Both client and middle tier applications can communicate directly with Analysis Services without using a provider. Client and middle tier applications may send XML for Analysis in SOAP packets over TCP/IP, HTTP, or HTTPS. The client may be coded using any language that supports SOAP. Communication in this case is most easily managed by Internet Information Services (IIS) using HTTP, although a direct connection to the server using TCP/IP may also be coded. This is the thinnest possible client solution for Analysis Services.

Projects

This solution has one project which contains a class that is used for the security proxy for SSAS.

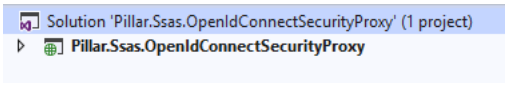


FIGURE 124 - PROJECTS VIEW

While reviewing the code, we found hardcoded urls. If there is any need to change or move the environment in the future, this url will need to be changed in the code and a new build will need to be created. More importantly is the fact that someone will have to remember that this url is in the code and not in a configuration file.

```
var req = (HttpWebRequest)WebRequest.Create("https://[redacted]uat-ssas.feisystems.com/msmdpump.dll");
```

We recommend moving this and all other hardcoded settings configuration files.

Architecture view for Pillar.Ssas.OpenIdConnectSecurityProxy

This is the simplest component in the FASAMS Application where it is the only project having a custom class in it.

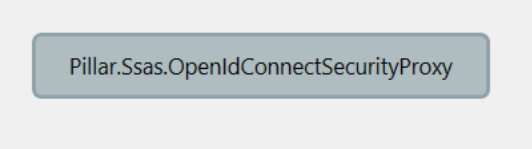


FIGURE 125 - ARCHITECTURE VIEW

11.2.2. Pillar.Ssrs.Extensions

This is a custom security extension for the SQL Server Reporting Services for FASAMS. Reporting Services provides an extensible architecture [6] that allows custom or forms-based authentication modules to be plugged in. Custom authentication extensions are developed when deployment requirements do not include Windows integrated security or Basic authentication. The most common scenario for using custom authentication is to support Internet or extranet access to a Web application. Replacing the default Windows Authentication extension with a custom authentication extension gives more control over how external users are granted access to the report server.

Deploying a custom authentication extension requires multiple steps that include copying assemblies and application files, modifying configuration files, and testing. This project uses a PowerShell script to copy DLL files. The script is part of the project and it is called CopyDllsForPackage.ps1 as highlighted in Figure 126 – Project View.

Custom authentication and authorization for Reporting Services is usually appropriate in the following scenarios:

- It is an Internet or extranet application that cannot use Windows accounts.
- Application uses custom-defined users and roles and need to provide a matching authorization scheme in Reporting Services.

As FASAMS uses custom-defined user and roles, this extension is needed to provide a uniform authentication mechanism to the users.

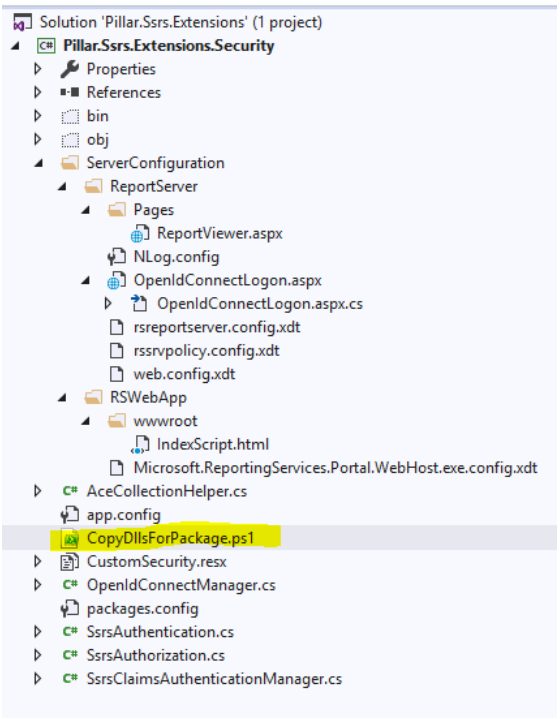


FIGURE 126 – PROJECT VIEW

Figure 127 - Class View shows classes in the project for custom security, SSRS Authorization and SSRS Authentication. Based on analysis of reference chains in the code, it is clear that this extension is only used for authentication and authorization in SSRS.

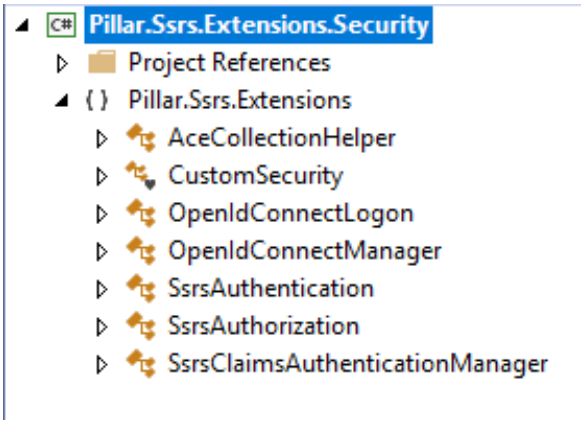


FIGURE 127 - CLASS VIEW

Projects

There is only 1 project within the Pillar.Ssrs.Extensions solution which has all the related source code for the custom security implementation for the FASAMS reporting services.

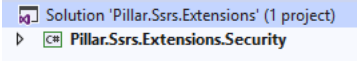


FIGURE 128 - PROJECTS VIEW

Figure 129 - Class View shows the classes for custom security, authentication, authorization and a few other classes in the project.

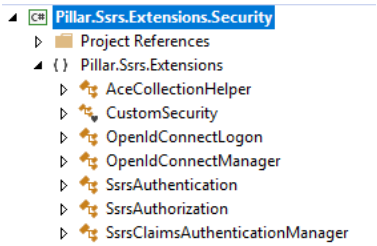


FIGURE 129 - CLASS VIEW

Architecture view for Pillar.Ssrs.Extensions

Below is the generic architecture [7] for SSRS Security Extension work flow. This demonstrates how authentication and authorization occurs using the default or custom security module. Note that FASAMS has a separate security database which stores the security related data, and the security extension authenticates against this database.

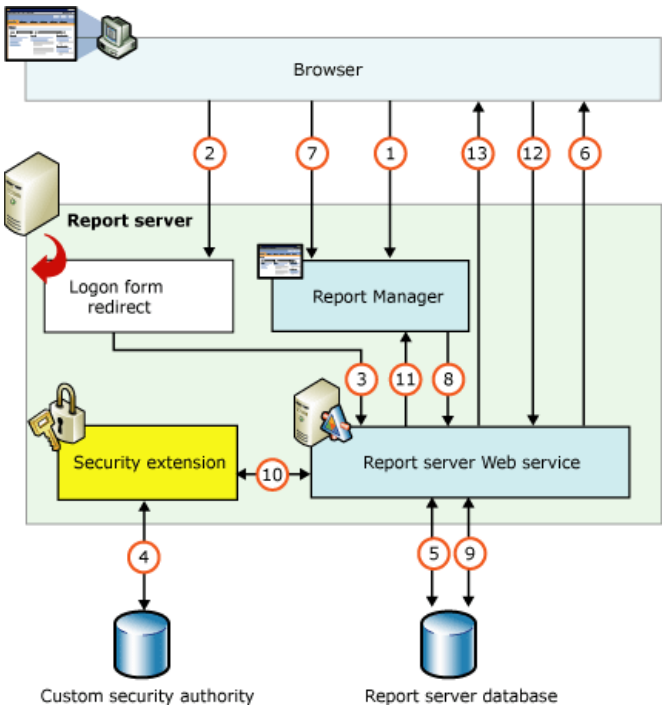


FIGURE 130 - SSRS SECURITY EXTENSION - EXAMPLE

1. A user tries to access the web portal by using a URL and is redirected to a form that collects user credentials for the client application.
2. The user submits credentials to the form.
3. The user credentials are submitted to the Reporting Services Web service through the LogonUser method.
4. The Web service calls the customer-supplied security extension and verifies that the user name and password exist in the custom security authority.
5. After authentication, the Web service creates an authentication ticket (known as a "cookie"), manages the ticket, and verifies the user's role for the Home page of the web portal.
6. The Web service returns the cookie to the browser and displays the appropriate user interface in the web portal.

7. After the user is authenticated, the browser makes requests to the web portal while transmitting the cookie in the HTTP header. These requests are in response to user actions within the web portal.
8. The cookie is transmitted in the HTTP header to the Web service along with the requested user operation.
9. The cookie is validated, and if it is valid, the report server returns the security descriptor and other information relating to the requested operation from the report server database.
10. If the cookie is valid, the report server makes a call to the security extension to check if the user is authorized to perform the specific operation.
11. If the user is authorized, the report server performs the requested operation and returns control to the caller.
12. After the user is authenticated, URL access to the report server uses the same cookie. The cookie is transmitted in the HTTP header.
13. The user continues to request operations on the report server until the session has ended.

The FASAMS project architecture diagram is found below.

FIGURE 131 - ARCHITECTURE VIEW

11.2.3. Bird.Portal.DataTier

This solution contains several projects which includes CLR, Data Tier, Reports and SSAS related modules. The majority of the projects are database related. Database objects written in managed code are called SQL Server Common Language Runtime (CLR) objects; constituent project Bird.Portal.Database.Clr is used for such objects. Bird.Portal.DataTier project is used for SQL server native objects and DDL scripts like tables, views, stored procedures etc. Bird.Portal.DynamicDataSets project only contains the reference for Bird.Portal.DataTier. Bird.Portal.Reports projects is the reporting project which contains all the SSRS reports (.rdl files) for FASAMS Reports module. The last project in the solution is Bird.Ssas.Model project which contains the Tabular Model for the analysis services objects.

Projects

One of the benefits of managed code is type safety, or the assurance that code accesses types only in well-defined, permissible ways. Before managed code is executed, the CLR verifies that the code is type safe. For example, the code is checked to ensure that no memory is read that has not previously been written. The CLR can also help ensure that code does not manipulate unmanaged memory.

The CLR provides a number of services to help achieve the design goals of CLR integration with SQL Server.

- Type safety verification
- Application domains
- Code Access Security (CAS)

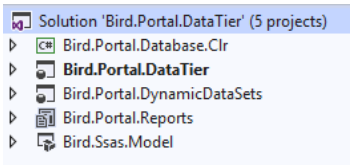


FIGURE 132 - PROJECTS VIEW

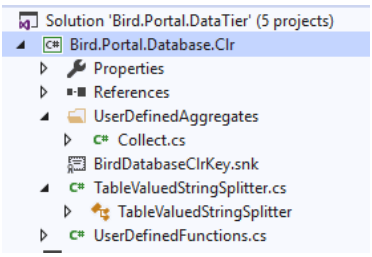


FIGURE 133 - PROJECTS/ CLASSES VIEW

Bird.Portal.DataTier project is used for the database object DDL scripts i.e. tables, stored procedures, views etc. There are publish profiles for multiple environments in the project. These can be used to deploy the database easily to DEV, UAT or PROD without making any changes to the configuration each time.

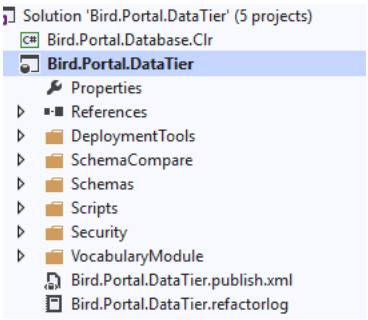


FIGURE 134 - PROJECT EXPANDED VIEW

Bird.Portal.DynamicDataSets project only contains the reference of Bird.Protal.DataTier assembly.

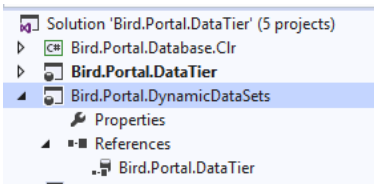


FIGURE 135 - PROJECT EXPANDED VIEW

Bird.Portal.Reports project is an SSRS project containing the .rdl files for the reporting module listed in FASAMS web application.

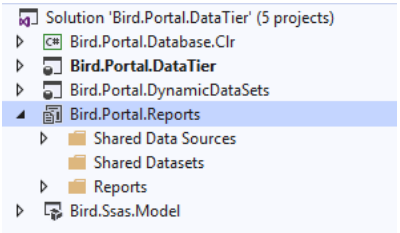


FIGURE 136 - PROJECT EXPANDED VIEW

Bird.Ssas.Model project contain tabular model for the analysis services.

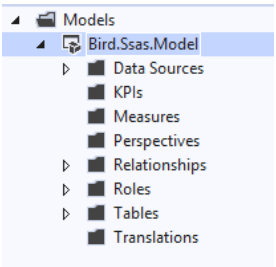


FIGURE 137 - PROJECT EXPANDED VIEW

Architecture view for Bird.Portal.DataTier

Bird.Portal.DataTier project is dependent on Bird.Portal.Database.Clr project whereas the other 3 projects are independent.

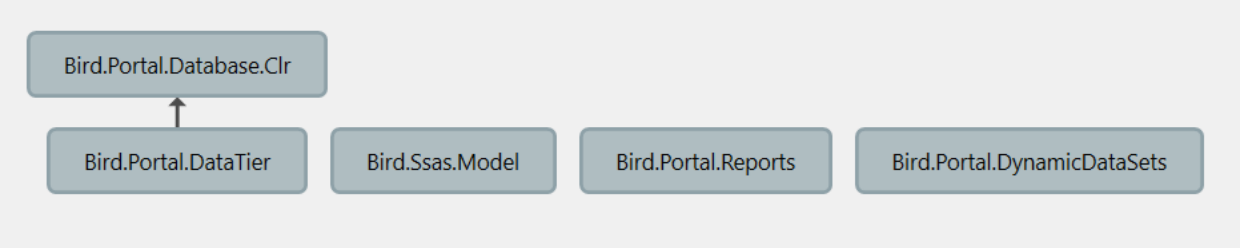


FIGURE 138 - ARCHITECTURE VIEW

11.2.4. Pillar.SftpServer

This solution has several projects. The main project in the solution is Pillar.SftpServer which handles SFTP transfers and parsing of loaded files for further processing. The solution also contains a few general-purpose projects that create library / dll files referenced by other solutions in the system. Some of these libraries contain business rule sets.

As shown in Figure 141 - Architecture View, the solution contains all of the Bird.Portal.* projects and FloridaDepartmentOfChildrenAndFamilies project. All of these reference the Pillar dll's.

It also has a windows service project and a console project. The windows service is the SFTP monitor and polls the SFTP site on configured schedule. It processes the files loaded by managed entities. The console application is just for unit testing this service.

Projects

The projects starting with the prefix of Bird.Portal and FloridaDepartmentOfChildrenAndFamilies are framework related projects in the solution that produce common binaries referenced by multiple applications. The main binary of this solution is Pillar.SftpServer, which contains the business logic for SFTP. It also has two other projects, one is Pillar.SftpServer.Console, which is a small project having a simple class and the second project is Pillar.SftpServer.WindowsService which is the entry point of this SFTP project. Windows services can be installed on windows operating systems which then keep running and monitoring. The service has a configured schedule to poll the SFTP site for uploaded files. When new files are found, they are downloaded and processed.

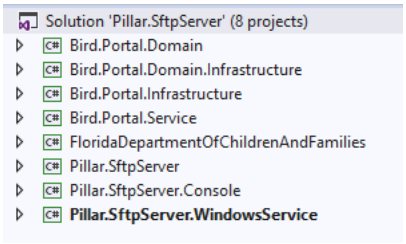


FIGURE 139 - PROJECTS VIEW

As shown in the highlighted item in the diagram below, Pillar.SftpServer implements standard FTP interface methods like Start(), Stop(), OnAuthentication() and OnFileUploaded().

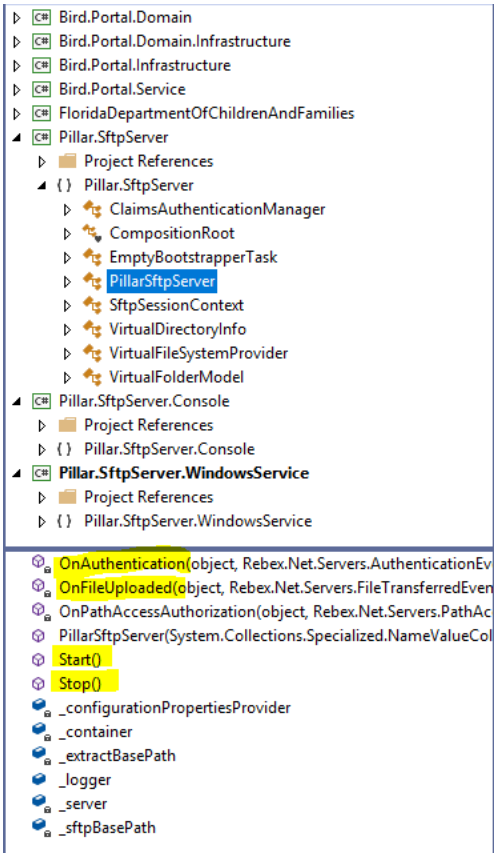


FIGURE 140 - PROJECTS/ CLASSES VIEW

Pillar.SftpServer.WindowsServer project logs its activities. This is a standard practice for any background processes, like an FTP monitoring service and helps with debugging. It is only coded for “Error” and “Debug” level logging though; best practices are to code for all verbosity levels (Fatal, Error, Warn, Info, Debug and Trace) and then control actual verbosity through configurations. The log is being written to a network path, which causes lower performance and it does not have auto purge and roll configured properly which can cause too much growth of the log files at higher verbosity levels.

Architecture view for Pillar.SftpServer

This architecture diagram of Pillar.SftpServer solution represents the hierarchy of different projects involved. The bottom two projects are the entry points for the Pillar.SftpServer project which has the business logic implemented

for the SFTP files and upload. However, the rest of the projects are framework projects producing generic libraries. These libraries are used by the SFTP service as well as other applications in the system.

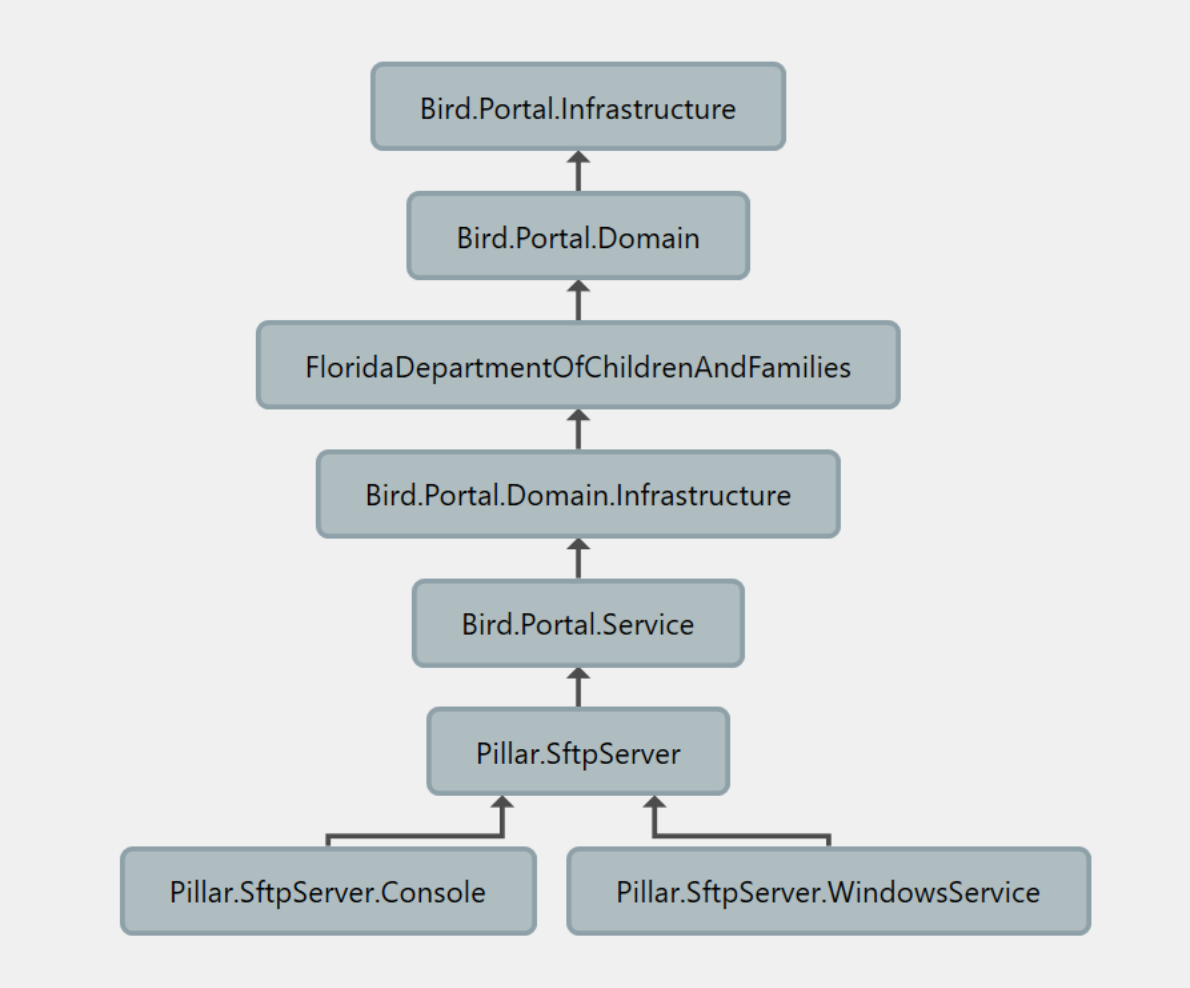


FIGURE 141 - ARCHITECTURE VIEW

Pillar.SftpServer is using the default port 22 for SFTP and it is hardcoded. Given that the SFTP's audience is limited to the 7 ME's, it may be a good idea to use a different port to improve security.

11.2.5. Pillar.TaskScheduler

Pillar.TaskScheduler solution has several projects. It has a few framework projects with naming pattern Bird.Portal.* and FloridaDepartmentOfChildrenAndFamilies. Other projects in this solution are used for FASAMS to perform different jobs which are under the task scheduler. Most of these projects are interfaces to specific systems like FSFN or IDS, that send /receive data to/from these systems and work as background processes through the job scheduler.

This solution's entry point projects are Pillar.TaskScheduler.Console and Pillar.TaskScheduler.WindowsService. Windows Service is the main controller that calls other interface projects to send/receive data.

Each of the interface projects has its specific configuration and schedule which is specified in the Pillar.TaskScheduler.WindowsService project configurations. As the interface binaries run from within the windows service process, the pick their configuration values from the windows service's configuration file.

Projects

There are 13 projects under this solution as shown in the diagram below. The following 6 projects are the interfaces for external systems to send/receive data and to perform business logic specific to the interfaced systems. All of these interfaces are controlled by the Pillar.TaskScheduler.WindowsService project which has respective configurations for each of them to start and connect with their target systems i.e. SFTP Credentials, Network Directories etc.

- FloridaDepartmentOfChildrenAndFamilies.FloridaSafeFamiliesNetwork
- FloridaDepartmentOfChildrenAndFamilies.InquiryDataSet
- FloridaDepartmentOfChildrenAndFamilies.MasterClientIndex
- FloridaDepartmentOfChildrenAndFamilies.MedicaidManagement
- FloridaDepartmentOfChildrenAndFamilies.ProviderLicensingAHCA
- FloridaDepartmentOfChildrenAndFamilies.ProviderLicensureDesignations

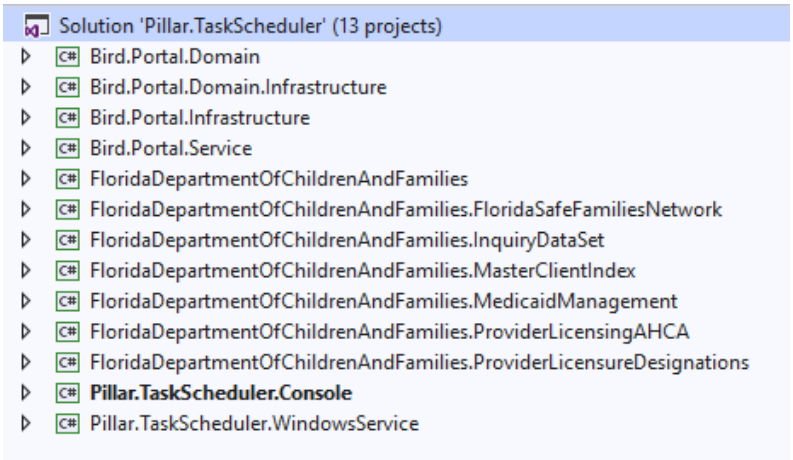


FIGURE 142 - PROJECTS VIEW

Architecture view for Pillar.TaskScheduler

Task Scheduler solution has 2 types of projects; one is for the framework libraries and the others are interfaces. The interfaces are controlled or called by the Windows Service project.

Following diagram shows dependency relationships among the service and framework projects.

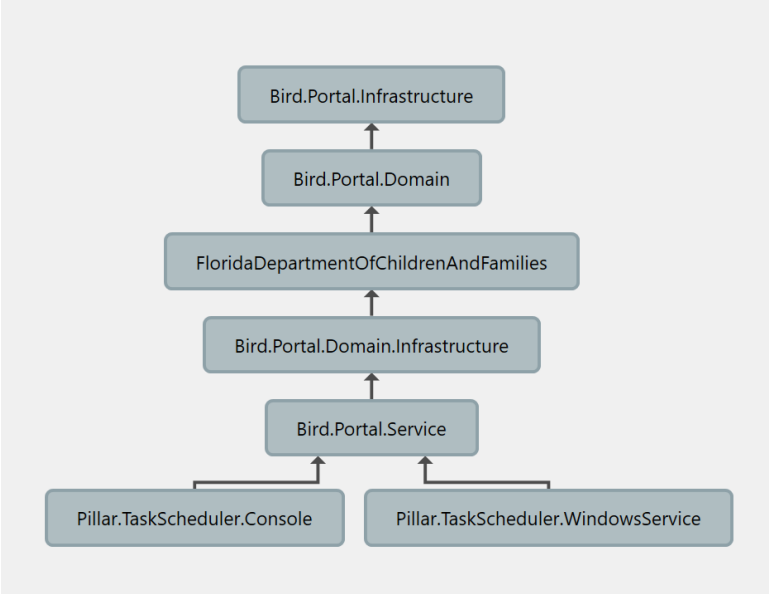


FIGURE 143 - ARCHITECTURE VIEW

Figure 144 - Projects Dependencies Graph shows dependency relationships among all the projects of this solution.

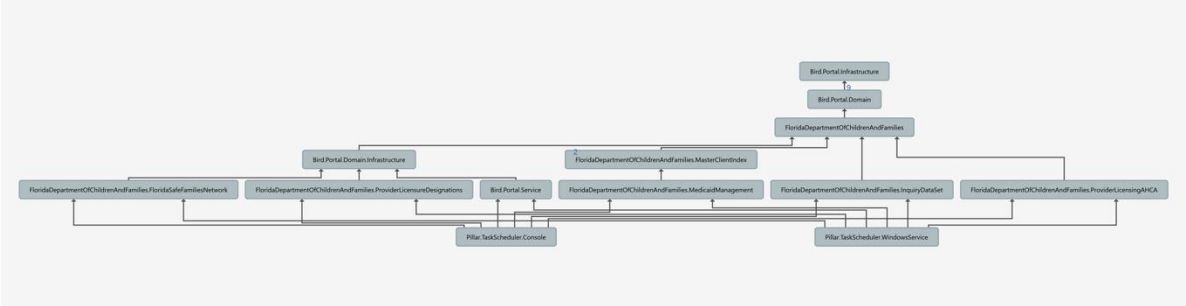


FIGURE 144 - PROJECTS DEPENDENCIES GRAPH

Figure 145 shows dependency relationships among the interface projects of this solution.

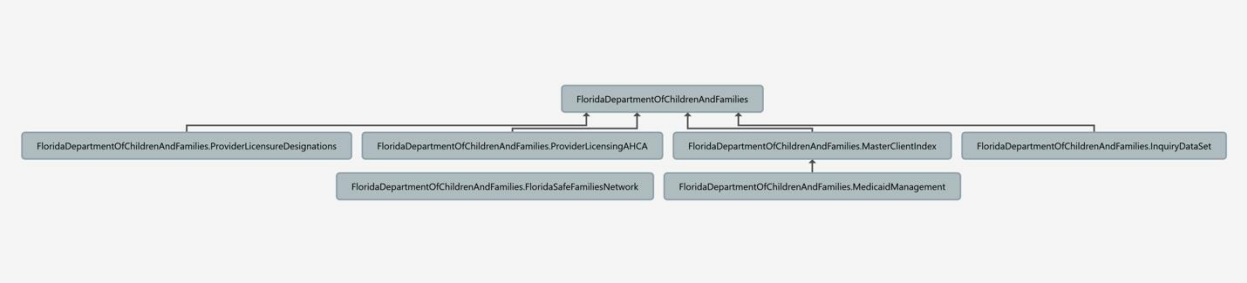


FIGURE 145 - FASAMS INTERFACE/ SERVICES DEPENDENCIES GRAPH

11.2.6. Pillar.SecurityServer

Pillar.SecurityServer solution contains several projects primarily related to security management for the FASAMS application. It has a few library projects and an MVC based project for the application security management. The

MVC based project is Pillar.SecurityServer and uses Angular, HTML, CSS etc. to provide the UI and business logic to manage the overall FASAMS project security. This project is also the entry point for the solution.

Projects

The Pillar.SecurityServer project is an MVC project for the security management of the application. It has client, role site, scope, security questions etc. related functionality.

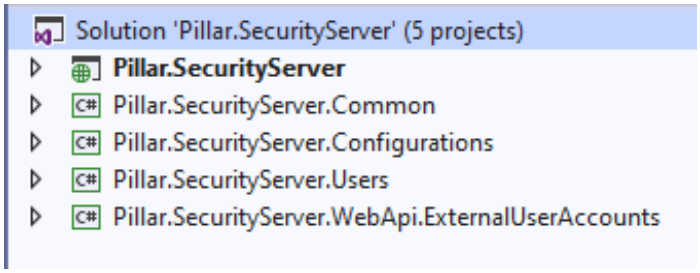


FIGURE 146 - PROJECTS VIEW

Architecture view for Pillar.SecurityServer

Figure 147 - Architecture View shows the dependency relationships for all projects of this solution.

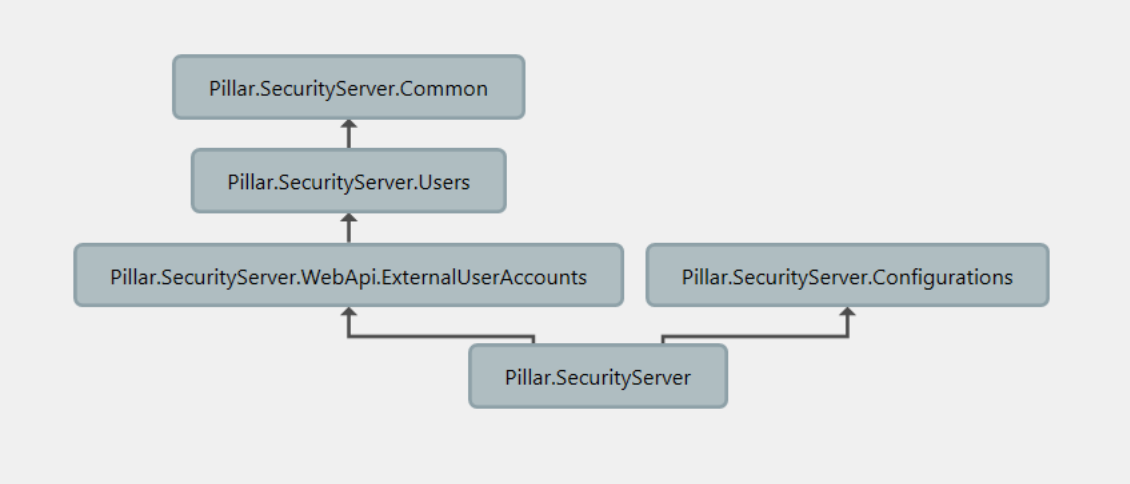


FIGURE 147 - ARCHITECTURE VIEW

11.2.7. Bird.Portal

Bird.Portal solution is the main entry point FASAMS application where users can interact with various FASAMS artifacts like Submissions, Reports etc. This solution has several projects which are categorized as Customer related and Tools. Bird.Portal.Web project is the entry point for FASAMS. The solution also contains some automated unit test projects.

The projects in this solution use different patterns and technologies; these include Type Script, Angular JS, UnitofWork Patterns, Karma Runner etc.

Microsoft TypeScript [8] is an open-source programming language developed and maintained by Microsoft. It is a strict syntactical superset of JavaScript and adds optional static typing to the language. ... There are third-party header files for popular libraries such as jQuery, MongoDB, and D3.js.

AngularJS [9] is a JavaScript-based open-source front-end web framework mainly maintained by Google and by a community of individuals and corporations to address many of the challenges encountered in developing single-page applications. It aims to simplify both the development and the testing of such applications by providing a framework for client-side model–view–controller (MVC) and model–view–view-model (MVVM) architectures, along with components commonly used in rich Internet applications. (This flexibility has led to the acronym MVW, which stands for "model-view-whatever" and may also encompass model–view–presenter and model–view–adapter.) In 2014, the original AngularJS team began working on the Angular web framework.

The repository and unit of work patterns [10] are intended to create an abstraction layer between the data access layer and the business logic layer of an application. Implementing these patterns can help insulate an application from changes in the data store and can facilitate automated unit testing.

The main goal for Karma [11] is to bring a productive testing environment to developers. The environment being one where they don't have to set up loads of configurations, but rather a place where developers can just write the code and get instant feedback from their tests.

Projects

There are 29 projects in Bird.Portal solution; some of these are class libraries and others are interfaces to communicate with external systems i.e. FSFN, IDS etc. The interfaces are grouped under the Customer folder in the solution. There are some general supporting projects as well, grouped under the Tools folder. The Customer folder also contains automated unit test projects.

In total, this solution has 20 class libraries, 7 automated unit tests, 1 console application and 1 web site project.

The entry point of the solution, Bird.Portal.Web, uses Angular.JS and TypeScript.

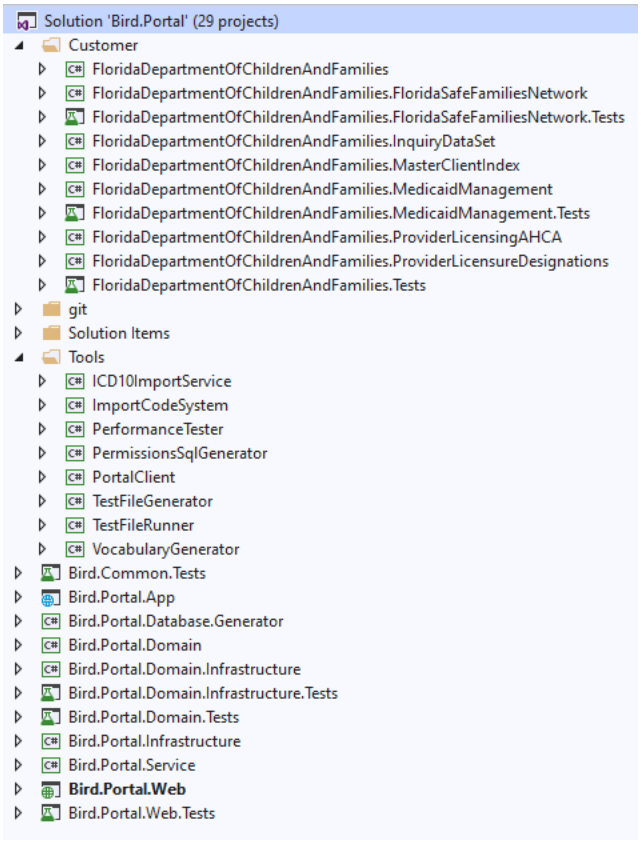


FIGURE 148 - PROJECTS VIEW

Architecture view for Bird.Portal

Figure 149 below shows dependency relationships for the main entry point of this solution: Bird.Portal.Web.

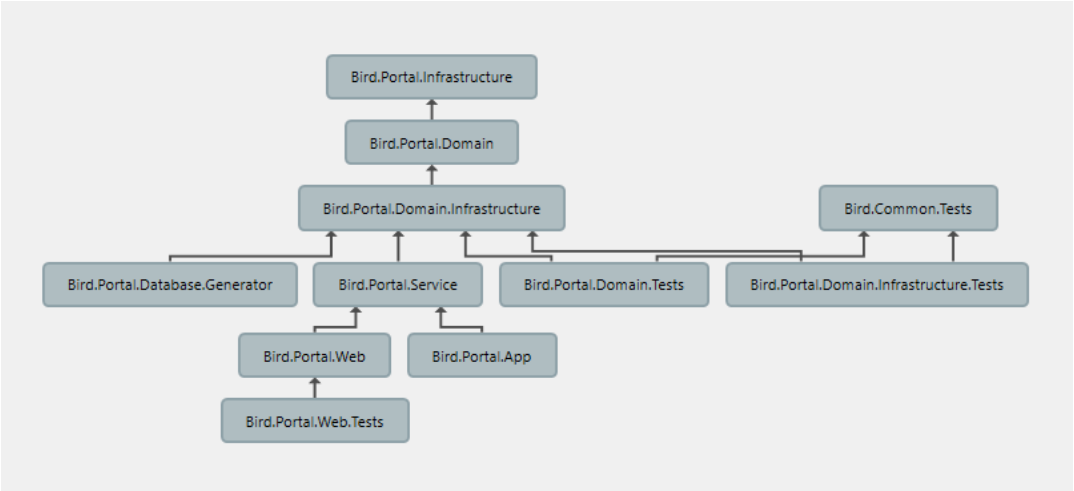


FIGURE 149 - ARCHITECTURE VIEW

Figure 150 below shows dependency relationships among all of the projects of this solution.

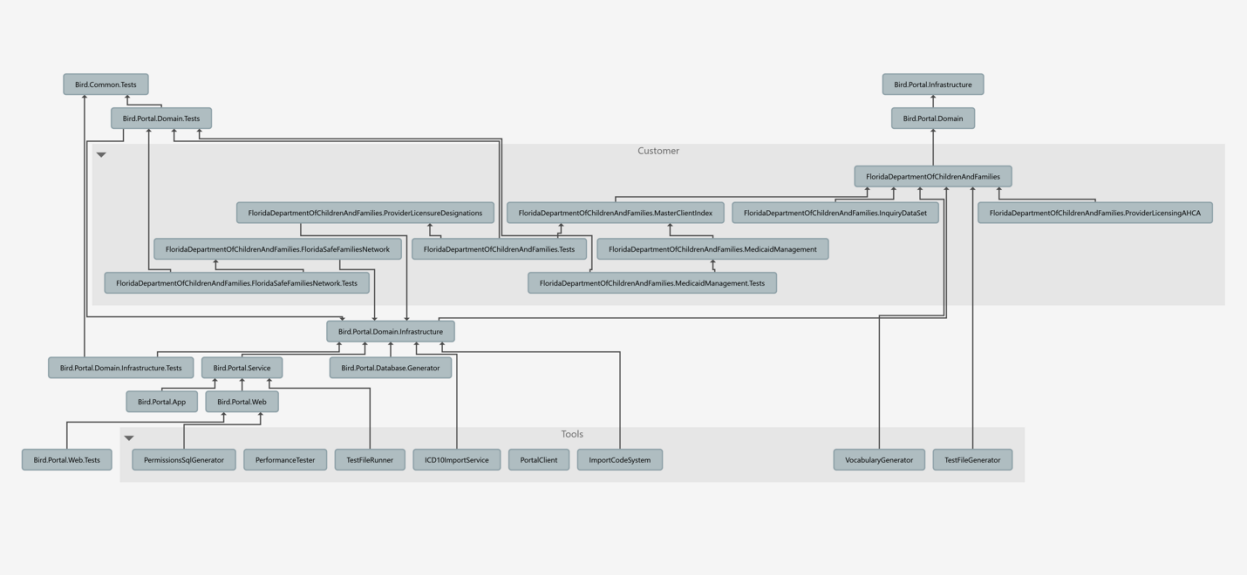


FIGURE 150 - BIRD.PORTAL PROJECTS DEPENDENCIES GRAPH

11.3. Pillar DLLs

FASAMS project is dependent on the following external DLLs which required for successful build and execution.

The current code references version 1.1.9 of these DLLs.

Name	Date modified	Type	Size	File version	Product version
Pillar.Common.dll	9/26/2018 6:19 PM	Application extension	64 KB	1.1.9.0	1.1.9
Pillar.Common.LightInject.dll	9/26/2018 6:19 PM	Application extension	16 KB	1.1.9.0	1.1.9
Pillar.Common.Tests.dll	9/26/2018 6:19 PM	Application extension	24 KB	1.1.9.0	1.1.9
Pillar.Domain.dll	9/26/2018 6:19 PM	Application extension	21 KB	1.1.9.0	1.1.9
Pillar.Domain.FluentRuleEngine.dll	9/26/2018 6:19 PM	Application extension	14 KB	1.1.9.0	1.1.9
Pillar.Domain.LightInject.dll	9/26/2018 6:19 PM	Application extension	7 KB	1.1.9.0	1.1.9
Pillar.FluentRuleEngine.dll	9/26/2018 6:19 PM	Application extension	87 KB	1.1.9.0	1.1.9
Pillar.FluentRuleEngine.LightInject.dll	9/26/2018 6:19 PM	Application extension	8 KB	1.1.9.0	1.1.9
Pillar.FluentRuleEngine.Tests.dll	9/26/2018 6:19 PM	Application extension	59 KB	1.1.9.0	1.1.9
Pillar.LightInject.dll	9/26/2018 6:19 PM	Application extension	109 KB	1.1.9.0	1.1.9
Pillar.LightInject.WebApi.dll	9/26/2018 6:19 PM	Application extension	114 KB	1.1.9.0	1.1.9
Pillar.Metadata.dll	9/26/2018 6:19 PM	Application extension	74 KB	1.1.9.0	1.1.9
Pillar.Metadata.LightInject.dll	9/26/2018 6:19 PM	Application extension	7 KB	1.1.9.0	1.1.9
Pillar.Nhibernate.dll	9/26/2018 6:19 PM	Application extension	55 KB	1.1.9.0	1.1.9
Pillar.Nhibernate.LightInject.dll	9/26/2018 6:19 PM	Application extension	8 KB	1.1.9.0	1.1.9
Pillar.NLog.dll	8/29/2018 1:57 AM	Application extension	10 KB	0.0.0.0	0.0.0.0
Pillar.QueryApi.dll	9/26/2018 6:20 PM	Application extension	49 KB	1.1.9.0	1.1.9
Pillar.QueryApi.LightInject.dll	9/26/2018 6:20 PM	Application extension	6 KB	1.1.9.0	1.1.9
Pillar.Security.dll	9/26/2018 6:20 PM	Application extension	19 KB	1.1.9.0	1.1.9
Pillar.Service.dll	9/26/2018 6:20 PM	Application extension	81 KB	1.1.9.0	1.1.9
Pillar.Service.LightInject.dll	9/26/2018 6:20 PM	Application extension	10 KB	1.1.9.0	1.1.9
Pillar.Ssrs.dll	9/26/2018 6:20 PM	Application extension	376 KB	1.1.9.0	1.1.9

FIGURE 151 - PILLAR DLLS

Below are the assembly names or DLLs for which there is no source code available:

1. Pillar.Common.dll
2. Pillar.Common.LightInject.dll

3. Pillar.Common.Tests.dll
4. Pillar.Domain.dll
5. Pillar.Domain.FluentRuleEngine.dll
6. Pillar.Domain.LightInject.dll
7. Pillar.FluentRuleEngine.dll
8. Pillar.FluentRuleEngine.LightInject.dll
9. Pillar.FluentRuleEngine.Tests.dll
10. Pillar.LightInject.dll
11. Pillar.LightInject.WebApi.dll
12. Pillar.Metadata.dll
13. Pillar.Metadata.LightInject.dll
14. Pillar.Nhibernate.dll
15. Pillar.Nhibernate.LightInject.dll
16. Pillar.NLog.dll
17. Pillar.QueryApi.dll
18. Pillar.QueryApi.LightInject.dll
19. Pillar.Security.dll
20. Pillar.Service.dll
21. Pillar.Service.LightInject.dll
22. Pillar.Ssrs.dll

11.4. References

1. https://docs.microsoft.com/en-us/visualstudio/ide/solutions-and-projects-in-visual-studio?view=vs-2019
2. https://support.microsoft.com/en-us/help/815065/what-is-a-dll
3. https://docs.microsoft.com/en-us/visualstudio/ide/managing-references-in-a-project?view=vs-2019
4. https://docs.microsoft.com/en-us/dotnet/framework/app-domains/assemblies-in-the-common-language-runtime
5. https://docs.microsoft.com/en-us/sql/analysis-services/multidimensional-models/olap-physical/client-architecture-requirements-for-analysis-services-development?view=sql-server-2017
6. https://docs.microsoft.com/en-us/sql/reporting-services/security/configure-custom-or-forms-authentication-on-the-report-server?view=sql-server-2017
7. https://docs.microsoft.com/en-us/sql/reporting-services/extensions/security-extension/security-extensions-overview?view=sql-server-2017
8. https://en.wikipedia.org/wiki/Microsoft_TypeScript
9. https://en.wikipedia.org/wiki/AngularJS
10. https://docs.microsoft.com/en-us/aspnet/mvc/overview/older-versions/getting-started-with-ef-5-using-mvc-4/implementing-the-repository-and-unit-of-work-patterns-in-an-asp-net-mvc-application
11. https://karma-runner.github.io/latest/index.html

12. AWS Review

12.1 Introduction

In this section of the document, the OZ cloud services team reviews the AWS FASAMS environment. The OZ cloud services team leverages the AWS Well-Architected Framework. The framework helps understand the pros and cons of decision one makes when building a system on AWS. The framework is based on five pillars:

Name	Description
Operational Excellence	The ability to run and monitor systems to deliver business value and to continually improve supporting processes and procedures.
Security	The ability to protect information, systems, and assets while delivering business value through risk assessments and mitigation strategies.
Reliability	The ability of a system to recover from infrastructure or service disruptions, dynamically acquire computing resources to meet demand, and mitigate disruptions such as misconfigurations or transient network issues.
Performance Efficiency	The ability to use computing resources efficiently to meet system requirements, and to maintain that efficiency as demand changes and technologies evolve.
Cost Optimization	The ability to run systems to deliver business value at the lowest price point.

The OZ cloud services team leverages the AWS well-Architected framework as our methodology to conduct AWS Infrastructure as a Service (IaaS) reviews. Additionally, in this section of the document, reference the diagram below (Figure.122) that was provided by the DCF team to the OZ team. Our recommendations and observations are based on the documentation, diagrams, and a conference call that occurred Friday, July 26, 2019 additional to the conference call the OZ team sent the FEI team some additional questions that were answer and received by the OZ team on August 7th, 2019.

12.2 Current State

Amazon Web Services (AWS) hosts the FASAMS application, FEI manages the management of the cloud infrastructure on behalf of DCF; FEI manages two subscriptions for DCF. The first subscription is a production/UAT subscription and a second subscription being train/test subscriptions. Both subscriptions are part of the US East 1 Region (Virginia).

FEI AWS management account is an AWS subscription that manages the following services:

- FASAMS EC2 instances
- Directory Services (AD)
- System Management
- FEI connects to the production/UAT and train/test subscription by having a peer to peer VPN tunnel.

The production/UAT subscription architecture is composed of the following components:

- Two availability groups
    - Enabling High-Availability
- Load-balancing
    - F5 – managed by DCF
    - Round-robin configuration
- Two EC2 instances for application server
    - Private-subnets
    - Security group enabled
    - EC2 AMI Windows server 2016
- Two EC2 instances for the database server
    - Private-subnets
    - Security group enabled.
    - A single database is running SQL 2016.
        - SSRS & SSAS - 2016

        - The database is always-on between Availability group 1 (AG1) and Availability group 2 (AG2).
            - During the conference call is was stated that a snap-shot of the production database is done and sent to a 3rd AWS subscription in the west region.
                - The data snapshot is sent to S3 storage and then sent to AWS glacier for archival. Note: the diagram does not depict that architecture.
    - Security / Monitoring / DevOps
        - Security
            - All users profiles must access FASAMS thru a VPN client (Aventail).
            - All internal users to DCF leverage active directory services for user authentication.
            - All external users to DCF leverage LDAP as their protocol of communication to request authentication.
            - Authorization to FASAMS is done at the application layer and not at the directory level.
            - Security groups separate all AWS components in the production subscriptions.
        - Monitoring
            - All logs generated by the application are sent to an AWS S3 bucket
                - Archived by AWS glacier services
            - All logs are analyzed by:
                - Cloudtrail
            - All alerts managed by CloudWatch
        - DevOps
            - The production environment is auditable via AWS config

The train/test subscription architecture is composed of the following components:

- One availability groups
- Load-balancing
    - F5 – managed by DCF
    - Round-robin configuration
- Three EC2 instances for application server
    - Train app
        - EC2 AMI Windows server 2016
    - Test app
        - EC2 AMI Windows server 2016
    - Pre app
        - EC2 AMI Windows server 2016
- One EC2 instance for the database server
    - A single database is running SQL 2016.
        - SSRS & SSAS - 2016 .
- Security / Monitoring / DevOps
    - Security
        - All users' profiles must access FASAMS thru a VPN client (Aventail).
        - All internal users to DCF leverage active directory services for user authentication.
        - All external users to DCF leverage LDAP as their protocol of communication to request authentication.
        - Authorization to FASAMS is done at the application layer and not at the directory level.
        - Security groups separate all AWS components in the production subscriptions.
    - Monitoring
        - All logs generated by the application are sent to an AWS S3 bucket
            - Archived by AWS glacier services
        - All logs are analyzed by:
            - Cloudtrail

- All alerts managed by CloudWatch
- DevOps
  - The production environment is auditable via AWS config

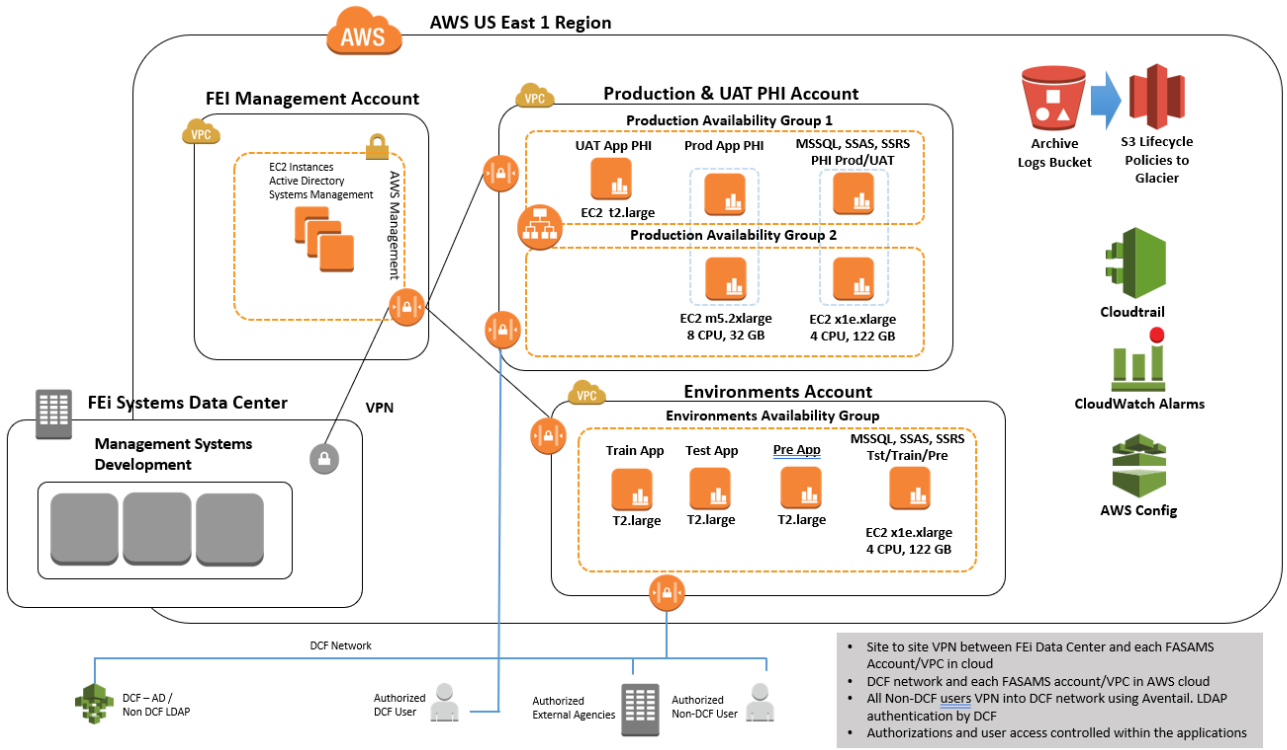


FIGURE 152- FASAMS AWS ENVIRONMENT

12.3 Observations and Recommendations

As stated above, the OZ cloud team leverages AWS Well-Architected framework as such; we provide our observations and recommendations thru the lenses of the five pillars.

Pillar	Observations	Recommendations
Operational Excellence	I. DCF/FEI are leveraging a variety of tools to monitor the performance of the environment. A combination of native AWS services and external services. Native services are AWS Cloudwatch, Config, and Cloudtrail in conjunction with external services such as Solarwinds and AlienVault.	I. No Recommendations at this time, The OZ cloud services team requires performance metric reports to provide a recommendation.

Security	I. The production subscription is hosted in the AWS east 1 region. II. Application handles authorization III. Management of AWS users	○ Due to the nature of the application and data sensitivity, our recommendation is to explore moving the production region to the AWS GovCloud East Region. The AWS GovCloud (US) Region maintains the region via U.S. citizens only and provides customers with the ability to access the region through FIPS 140-2 service endpoints. FEI provide the following answer as to why the AWS GovCloud region was discarded: "This was discussed with the AWS support team. The current region we are in supports all the security requirements. AWS recommends not using the GovCloud unless it comes under a federal DOD requirement for the GovCloud. ▪ Decision was based on discussion with AWS and were then discussed with DCF" I. Explore the possibility to remove the authorization from the application layer and centralize it as part of directory services by mapping the user's roles of the application to the directory services. II. It is a standards practice to ensure that all subject matter experts accessing AWS services must authenticate using Multi-factor authentication (MFA)
Reliability	I. Load balancing of traffic is handle outside AWS II. Application is highly available; however, it is not Disaster Recovery enabled. III. AWS Cloud-Formation Templates	○ It was said during the conference call that DCF handles all network/application traffic thru the use of F5 with a round-robin rule. During the call, there was no technical explanation on why was F5 leverage to handle traffic to the application. OZ recommendations are to explore if AWS ELB can be used in conjunction with the EC2 instances to ensure the application can meet additional customer demand spikes. FEI provided the following response to why AWS ELB was abandoned as a load-balancer: "FEI's original proposal was to use the AWS ELB but that changed during DDI

		▪ Email documentation shows FEI requested that DCF make a decision between F5 and ELB ▪ Ronnie conveyed to Chad that Cale wanted all traffic to go through the DCF VIP • This was the reason for using the DCF VIP with F5 • FEI was unaware of the DCF F5 load balancer until it was requested to be used by DCF " I. Depending on the business priority and revenue generation of the application, it is our recommendation to explore replicating the application in a west region. Exploration of using Cloudfront could be a possible manner to keep the cost down. II. If point III, becomes cost prohibited, then it is our recommendation to ensure that the current production deployment is saved as an AWS Cloud-Formation template.
Performance Efficiency	I. EC2 auto-scaling seems to be unused II. Application database serves as both the operational, transactional database and the reporting database	As was said in point I of the category above. EC2 auto-scaling must be configured to be able to scale up/down the EC2 instance automatically. Add one additional instance of EC2 on AG1 and create a reporting database. Follow the same approach for AG2.
Cost Optimization	Re-architecture of database Management of database	Explore the use of replacing MS SQL with either AWS Aurora or MySQL. This may bring cost reduction on Microsoft licensing. SSAS and SSRS can also be replaced with AWS services such as Athena and Quickinsight Explore the use of amazon RDS for database management. RDS can simply the amount of time the dbas spend on database management.

Appendices

Appendix A – Terms and Definitions

Term	Definition
508	Section 508, an amendment to the United States Workforce Rehabilitation Act of 1973, is a federal law mandating that all electronic and information technology developed, procured, maintained, or used by the federal government be accessible to people with disabilities.
ACCESS	Automated Community Connection to Economic Self-Sufficiency
AHCA	Agency for Health Care Administration
API	Application Programming Interface
CARS	Contract Accountability and Reporting System
CSV	Comma-Separated Values
DDI	Design, Development, & Implementation
FASAMS	Financial and Services Accountability Management System
FMMIS	Florida Medicaid Management Information System
FSFN	Florida Safe Families Network
ICD	International Classification of Diseases
IDS	Inquiry Data Set
JAD	Joint application design (JAD) is a process used in the life cycle area of the dynamic systems development method (DSDM) to collect business requirements while developing new information systems for a company.
MCI	Master Client Index
ODS	An operational data store (ODS) is a database designed to integrate data from multiple sources for additional operations on the data, for reporting, controls and operational decision support.
PDF	Portable Document Format
PLADS	Provider Licensing and Designation System
RDM	Record Data Model Specification Document

Term	Definition
SAMHSA	The Substance Abuse and Mental Health Services Administration (SAMHSA) is a branch of the U.S. Department of Health and Human Services. It is charged with improving the quality and availability of treatment and rehabilitative services in order to reduce illness, death, disability, and the cost to society resulting from substance abuse and mental illnesses.
SMMC	Statewide Medicaid Managed Care (SMMC) – Managed Medical Assistance Program
SSAS	SQL Server Analysis Services (SSAS) is the technology from the Microsoft Business Intelligence stack, to develop Online Analytical Processing (OLAP) solutions. In simple terms, the user can use SSAS to create cubes using data from data marts / data warehouse for deeper and faster data analysis.
SSRS	SQL Server Reporting Services (SSRS) is a server-based report generating software system from Microsoft.
SSAS Tabular Model	Tabular models are Analysis Services databases that run in-memory or in DirectQuery mode, accessing data directly from backend relational data sources.
SFTP	SFTP (SSH File Transfer Protocol) is a secure file transfer protocol. It runs over the SSH protocol. It supports the full security and authentication functionality of SSH.
Staging	This term will refer to a series of internal staging tables that FASAMS uses to keep track of each submission. Staging data represents what was submitted on the data sets, not what ultimately got stored in the FASAMS ODS database.
TEDS	Treatment Episode Data Set. SAMHSA's standard dataset to report substance abuse and mental health admissions, discharges, and updates.
URL	A Uniform Resource Locator (URL), colloquially termed a web address, [1] is a reference to a web resource that specifies its location on a computer network and a mechanism for retrieving it. A URL is a specific type of Uniform Resource Identifier (URI), [2] although many people use the two terms interchangeably.
XML	In computing, Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.
RDL	Report Definition Language
Solution	A project is contained within a solution [1]. Despite its name, a solution is not an "answer". It's simply a container for one or more related projects, along with build information, Visual Studio window settings, and any miscellaneous files that aren't associated with a particular project. A solution is described by a text file (extension .sln) with its own unique format; it's not intended to be edited by hand.

Term	Definition
Project	When you create an app, website, plug-in, etc. in Visual Studio, you start with a project. In a logical sense, a project contains all the source code files, icons, images, data files, etc. that are compiled into an executable, library, or website. A project also contains compiler settings and other configuration files that might be needed by various services or components that your program communicates with. A project is defined in an XML file with an extension such as .csproj. This file contains a virtual folder hierarchy, and paths to all the items in the project. It also contains the build settings.
DLL	A DLL [2] is a library that contains code and data that can be used by more than one program at the same time.
References	A reference [3] is essentially an entry in a project file that contains the information that Visual Studio needs to locate the component or the service.
Assembly	Assemblies [4] are the building blocks of .NET Framework applications; they form the fundamental unit of deployment, version control, reuse, activation scoping, and security permissions. An assembly is a collection of types and resources that are built to work together and form a logical unit of functionality.

Appendix B - Supporting Documents & References

Name	Link/Location	Description
Pamphlet 155-2	Current Version 12: http://www.myflfamilies.com/service-programs/substance-abuse/pamphlet-155-2-v12 New FASAMS version: FEI Systems Portal > FEI Collaboration > FASAMS > Shared Docs > Technical Documentation > Published Chapters	The DCF Pamphlet 155-2 website contains several documents, organized into chapters, for each of the data sets, and supporting documentation. FASAMS published chapters will eventually become the next version of the pamphlet.
Pamphlet 155-2, Introduction	https://www.myflfamilies.com/service-programs/samh/155-2/155-2-v13/Chapter%2001%20Introduction.pdf	This pamphlet specifies the data files and file layout requirements for collecting and reporting data on persons served.

Name	Link/Location	Description
Record Data Model Specification Document (RDM)	It is available on the FASAMS procurement site, and directly, here:http://www.dcf.state.fl.us/programs/samh/FASAMS/library/Record%20Data%20Model%20Doc/FASAMS-ITN03U17GN1-Record-Data-Model-Specification-Document.pdf	Supports the definition of business requirements for the FASAMS by defining the data needed and created by DCF Substance Abuse and Mental Health (SAMH) business processes, and to define the interactions between users and the system and data.
Dynamic Data Sets Functional Requirements Document	FEI Systems Portal > FEI Collaboration > FASAMS > Shared Docs > Requirements > Analysis > Dynamic Data Sets FASAMS Functional Design Document - Dynamic Datasets.docx	This document describes the Dynamic Data Sets business functional and technical requirements.
FSFN Extract Functional Requirements Document	FEI Systems Portal > FEI Collaboration > FASAMS > Shared Docs > Requirements > Analysis > Interfaces > Child Welfare (FSFN) FASAMS Functional Design Document - FSFN Extract.docx	This document describes the FSFN Extract business and functional and technical requirements.
Client Search Report Requirements Document	FEI Systems Portal > FEI Collaboration > FASAMS > Shared Docs > Requirements > Report Documentation Client Search Report.docx	This document describes the Client Search Report requirements.
Provider Expenditure Validation Report – Changes for Balance and Carry Forward Expenditures Report Requirements Document	FEI Systems Portal > FEI Collaboration > FASAMS > Shared Docs > Requirements > Report Documentation Provider Expenditure Validation Report - Changes for Balance and Carry Forward Expenditures.docx	This document describes the Provider Expenditure Validation Report requirements, specifically for the addition of the columns for the OCA Balance and Carry Forward information.
Acute Care Report Requirements Document	FEI Systems Portal > FEI Collaboration > FASAMS > Shared	This document describes the Acute Care Report requirements.

Name	Link/Location	Description
	Docs > Requirements > Report Documentation Acute Care Reports.docx	
Contract Compliance Report Requirements Document	FEI Systems Portal > FEI Collaboration > FASAMS > Shared Docs > Requirements > Report Documentation Contract Compliance Reports.docx	This document describes the Contract Compliance Reports requirements.
Historical Data Rule Changes Functional Requirements Document	FEI Systems Portal > FEI Collaboration > FASAMS > Shared Docs > Requirements > Analysis > HistoricalDataImplications FASAMS Functional Design Document - Historical Data Rule Changes.docx	This document describes the Historical Data Rule Changes functional requirements.
Vocabulary Effective and Expiration Date Functional Requirements Document	FEI Systems Portal > FEI Collaboration > FASAMS > Shared Docs > Requirements > Analysis > Vocabulary FASAMS Functional Design Document - Vocabulary Effective and Expiration Date.docx	This document describes the Vocabulary Effective and Expiration Date business and functional requirements.
FASAMS Technical Architecture	FEI Systems Portal > FEi Collaboration > FASAMS > Shared Docs > DDI > Technical Documentation > Technical Architecture.pptx	This document describes the FASAMS technical architecture
Network Server Diagrams	FEI Systems Portal > FEi Collaboration > FASAMS > Shared Docs > DDI > Technical Documentation > Network Server Diagrams.vsdx	This document describes the network/server configuration

Appendix C – Comparison of RTVM Document and the Functional Design Document

RTVM - full	Requirement (The system will:)	Doc Ref 1 Reference Location / Requirement ID #	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Location in FDD	Location in FDD - Stage 1	Location in FDD - Stage 2	Location in FDD - Stage 3	Location in FDD - Stage 4
00.00.0008	The system shall provide the capability to use national standard code sets (e.g., ICD-10, CPT, HCPCS).	9.8	9.9	9.8	N/A	Not present	not present	9.8	9.8	not present	not present	not present
00.00.0010	The system shall provide error codes and code descriptions explaining the reason a submitted record is being rejected (e.g., invalid OCA code, invalid admission date).	9.11	9.15	9.11	N/A	Not present	not present	9.11	9.11	not present	not present	not present
00.05.0077	The system shall provide the ability to perform secure file transfers using a file transfer method (e.g., SFTP, FTPS, SSH).	9.24	9.4	9.4	N/A	Not present	not present	9.4, 9.24	9.4	9.24	not present	not present
00.06.0057	The system shall provide non-technical specifications for each algorithm.		N/A	N/A	N/A	Not present	field empty	not present	Not present	not present	not present	not present

RTVM - full	Requirement (The system will:)	Doc Ref 1 Reference Location / Requirement ID #	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Location in FDD	Location in FDD - Stage 1	Location in FDD - Stage 2	Location in FDD - Stage 3	Location in FDD - Stage 4
00.06.0067	The system shall provide a selectable list of attributes to create an ad hoc report in the FASAMS user interface.	9.21	N/A	9.21	N/A	9.21	not present	not present	Not present	not present	not present	not present
00.06.0069	The system shall allow the user to select time frames (e.g., start date/time, end date/time) for report results in the FASAMS user interface.	9.41 9.42	N/A	N/A	N/A	9.41 9.42	not present	not present	Not present	not present	not present	not present
01.02.0017	The system shall provide a capability to associate the admission record to the contract (contractor ID, sub-contractor ID) as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation)	9.33	N/A	N/A	9.33	Not present	not present	9.33	Not present	not present	9.33	not present

RTVM - full	Requirement (The system will:)	Doc Ref 1 Reference Location / Requirement ID #	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Location in FDD	Location in FDD - Stage 1	Location in FDD - Stage 2	Location in FDD - Stage 3	Location in FDD - Stage 4
01.02.0018	The system shall automatically calculate the target population for a client as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	9.8	9.9	9.8	N/A	Not present	not present	9.8	9.8	not present	not present	not present
01.04.0014	The system shall provide the capability to track clients awaiting substance abuse and mental health services by level of care as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	9.18	N/A	9.18	N/A	Not present	not present	9.18	Not present	9.18	not present	not present

RTVM - full	Requirement (The system will:)	Doc Ref 1 Reference Location / Requirement ID #	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Location in FDD	Location in FDD - Stage 1	Location in FDD - Stage 2	Location in FDD - Stage 3	Location in FDD - Stage 4
01.05.0022	The system shall be configurable to accept a standard formatted record of acute care (i.e., CSU, Detox, ARF) treatment episode data per client as defined in the Pamphlet 155-2 (See Pamphlet 155-2 in Supporting Documents).	9.19	N/A	9.19	N/A	Not present	not present	not present	Not present	not present	not present	not present
01.05.0025	The system shall provide the capability to associate Child Welfare involvement information to client-specific service events.	9.38	N/A	N/A	N/A	9.38	not present	not present	Not present	not present	not present	not present
01.05.0033	The system shall track the length of stay on a given level of care under an admission, within an episode of care.	9.8	9.9	9.8	N/A	Not present	not present	9.8	9.8	not present	not present	not present
02.10.0051	The system shall provide the capability to track utilization of set aside funds.	9.17	N/A	N/A	9.17	Not present	not present	not present	Not present	not present	not present	not present

RTVM - full	Requirement (The system will:)	Doc Ref 1 Reference Location / Requirement ID #	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Location in FDD	Location in FDD - Stage 1	Location in FDD - Stage 2	Location in FDD - Stage 3	Location in FDD - Stage 4
02.10.0055	The system shall calculate the difference between the contracted rate for services for the ME and the Provider.	9.32	N/A	N/A	N/A	9.32	not present	9.32	Not present	not present	9.32	not present
02.11.0042	The system shall be configurable to accept nightly updates of the annual budget allocations from the DCF Budget Ledger System.	9.27	N/A	N/A	9.27	Not present	not present	9.27	Not present	not present	9.27	not present
02.12.0074	The system shall provide the capability to accept ACHA data to determine a client's funding eligibility (e.g., Medicare, Medicaid, managed care plans, and third party insurance).	9.29	N/A	N/A	N/A	Not present	9.29	9.29	Not present	not present	9.29	not present

RTVM - full	Requirement (The system will:)	Doc Ref 1 Reference Location / Requirement ID #	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Location in FDD	Location in FDD - Stage 1	Location in FDD - Stage 2	Location in FDD - Stage 3	Location in FDD - Stage 4
02.13.0064	The system shall be configurable to track payment categories (e.g., Capitated, Cost Reimbursement, Bundled Rates, Capacity, Utilization) as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	9.20	N/A	N/A	9.20	Not present	not present	not present	Not present	not present	not present	not present
03.00.0081	The system shall produce standard reports as defined by DCF (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).		N/A	N/A	N/A	field empty	field empty	not present	Not present	not present	not present	not present
no number	The system shall have a way to manage the allowed values for fields in FASAMS.	9.15	9.19	9.15	N/A	Not present	not present	9.15	9.15	not present	not present	not present

RTVM - full	Requirement (The system will:)	Doc Ref 1 Reference Location / Requirement ID #	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Location in FDD	Location in FDD - Stage 1	Location in FDD - Stage 2	Location in FDD - Stage 3	Location in FDD - Stage 4
no number	For substance abuse providers, the FEIN for ProviderID must exist in the DCF Provider Licensure and Designations System (PLADS).	9.34	N/A	N/A	9.34	Not present	not present	9.34	Not present	not present	9.34	not present

Appendix D – Map RTVM to Test Plans

RTVM - full	Requirement (The system will:)	Doc Ref - Test Plan Reference Location / Test Case Identifier	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Test case number - map to UAT Results and Integration and system test results (excel) - column A	Test case number - map to UAT Results and Integration and system test results (excel) - column B	Column D - UAT results (excel)	Column E - UAT results (excel)
00.00.0008	The system shall provide the capability to use national standard code sets (e.g., ICD-10, CPT, HCPCS).	5616 5617 5618	9.9	9.8	N/A	Not present	not present	not present	5616 -present - not sure if it tests this 5618 - present to set values, and not delete rules	not present	not present
00.00.0010	The system shall provide error codes and code descriptions explaining the reason a submitted record is being rejected (e.g., invalid OCA code, invalid admission date).	5150, 6322, 6323, 6324, 6325, 6593	9.15	9.11	N/A	Not present	not present	not present	not present	Vocabulary tab	not present
00.05.0077	The system shall provide the ability to perform secure file transfers using a file transfer method (e.g., SFTP, FTPS, SSH).	4607, 7560, 7561, 7663, 7564, 7590	9.4	9.4	N/A	Not present	not present	not present	not present	not present	not present

RTVM - full	Requirement (The system will:)	Doc Ref - Test Plan Reference Location / Test Case Identifier	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Test case number - map to UAT Results and Integration and system test results (excel) - column A	Test case number - map to UAT Results and Integration and system test results (excel) - column B	Column D - UAT results (excel)	Column E - UAT results (excel)
00.06.0057	The system shall provide non-technical specifications for each algorithm.	7860, 8000, 7999, 8001, 8002, 8025, 8029, 8044, 7288, 7295, 7296, 7297, 7301, 7302, 7303, 7307, 7308, 7309, 7337, 7338, 7339, 7643, 7644, 7645, 7649, 7651, 7653, 7652, 7848, 7578, 7580, 7581, 7583, 7587, 7588, 7589, 7601, 7604, 7606, 7608, 7851, 6787, 6789, 6875, 6876, 6877, 6879, 6790, 6791, 6818, 6838, 6839, 6865, 6867,6868, 6869, 7624, 7625, 7626, 7627, 7628, 7629,7630, 7631, 7632, 7633, 7634, 7635, 7702, 7636, 7637, 7638, 7639, 7640, 8112, 8114, 8115, 8132, 8128, 8129, 8130, 8133, 7288, 7295, 7296, 7297, 7301, 7302, 7303, 7307, 7308, 7309, 7337, 7338, 7339	N/A	N/A	N/A	Not present	field empty	no value	not present	not present	not present

RTVM - full	Requirement (The system will:)	Doc Ref - Test Plan Reference Location / Test Case Identifier	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Test case number - map to UAT Results and Integration and system test results (excel) - column A	Test case number - map to UAT Results and Integration and system test results (excel) - column B	Column D - UAT results (excel)	Column E - UAT results (excel)
00.06.0067	The system shall provide a selectable list of attributes to create an ad hoc report in the FASAMS user interface.	6485, 7447, 7449, 7465, 7466, 7467, 7468, 7469, 7470, 7450, 7451, 7452, 7453 5090 5118 5121 5188 5191 5182 5196	N/A	9.21	N/A	9.21	not present	found 7465, 7466, 7467, 7468, 7469, 7470, 7450, 7452, does not look like testing this requirement	6485 - seems to be related but not completely	SQL server report tab	
00.06.0069	The system shall allow the user to select time frames (e.g., start date/time, end date/time) for report results in the FASAMS user interface.	6095, 6909, 7148, 7224, 7371,	N/A	N/A	N/A	9.41 9.42	not present	6095, 6909, 7148 - all date filters 7224, 7371 - does not seem to be related	not present	no number	no number
01.02.0017	The system shall provide a capability to associate the admission record to the contract (contractor ID, sub-contractor ID) as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation)	5242, 8753, 8762, 8774, 8802, 8892, 8897	N/A	N/A	9.33	Not present	not present	not present	not present	no number	no number

RTVM - full	Requirement (The system will:)	Doc Ref - Test Plan Reference Location / Test Case Identifier	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Test case number - map to UAT Results and Integration and system test results (excel) - column A	Test case number - map to UAT Results and Integration and system test results (excel) - column B	Column D - UAT results (excel)	Column E - UAT results (excel)
01.02.0018	The system shall automatically calculate the target population for a client as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	5084, 6537, 6538, 6539, 6540, 6541, 6542, 6543, 6544, 6545, 6546, 6547, 6548, 6549, 6549, 6550, 6551, 6552, 6555, 7498, 7504, 7542, 7565, 7572, 7573, 7579, 7583, 7584, 7585, 7588, 7599, 7601, 7602, 7607, 7608, 7613, 7616	9.9	9.8	N/A	Not present	not present	not present	not present	not present	not present
01.04.0014	The system shall provide the capability to track clients awaiting substance abuse and mental health services by level of care as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	5173, 7237, 7238, 7239, 7240, 7251, 7254, 7257, 7258, 7259, 7260	N/A	9.18	N/A	Not present	not present	7257, 7259 present but not related to this description	not present	no number	not present

RTVM - full	Requirement (The system will:)	Doc Ref - Test Plan Reference Location / Test Case Identifier	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Test case number - map to UAT Results and Integration and system test results (excel) - column A	Test case number - map to UAT Results and Integration and system test results (excel) - column B	Column D - UAT results (excel)	Column E - UAT results (excel)
01.05.0022	The system shall be configurable to accept a standard formatted record of acute care (i.e., CSU, Detox, ARF) treatment episode data per client as defined in the Pamphlet 155-2 (See Pamphlet 155-2 in Supporting Documents).	5174, 7659, 7178, 7179,7180, 7645, 7646, 7654, 7655, 7656, 7657, 7659, 7660, 7661, 7663, 7664, 7665, 7666, 7667, 7668, 7669, 7670, 7671, 7672, 7673, 7674, 7675, 7676, 7677, 7678, 7679, 7806, 7807, 7808, 7809, 7810, 7812, 7829, 7831, 7830, 7853, 7854, 7855, 7856, 7857, 7858, 7861, 7861, 7863, 7874, 7884, 7892	N/A	9.19	N/A	Not present	not present	not present	not present	no number	not present
01.05.0025	The system shall provide the capability to associate Child Welfare involvement information to client-specific service events.	7280, 7289, 7311, 7345, 7370	N/A	N/A	N/A	9.38	not present	7280, 7370present but not related to this description 7289, 7345 - similar but seem to be referring to an extract file	not present	no number	no number
01.05.0033	The system shall track the length of stay on a given level of care under an admission, within an episode of care.	5084, 6537, 6538, 6539, 6540, 6541, 6542, 6543, 6544, 6545, 6546, 6547, 6548, 6549, 6549, 6550, 6551, 6552, 6555, 7498, 7504, 7542, 7565, 7572, 7573, 7579, 7583, 7584, 7585, 7588, 7599, 7601, 7602, 7607, 7608, 7613, 7616	9.9	9.8	N/A	Not present	not present	not present	not present	not present	not present

RTVM - full	Requirement (The system will:)	Doc Ref - Test Plan Reference Location / Test Case Identifier	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Test case number - map to UAT Results and Integration and system test results (excel) - column A	Test case number - map to UAT Results and Integration and system test results (excel) - column B	Column D - UAT results (excel)	Column E - UAT results (excel)
02.10.0051	The system shall provide the capability to track utilization of set aside funds.	5172, 8398, 8399, 8594, 8400, 8401, 8623, 8623, 8621, 8402, 8624, 8625, 8426, 8429, 8403, 8404, 8425, 8405, 8406, 8407, 8408, 8585, 8588, 8617, 8618	N/A	N/A	9.17	Not present	not present	not present	not present	no number	no number
02.10.0055	The system shall calculate the difference between the contracted rate for services for the ME and the Provider.	4176, 6542, 6544, 6562, 6564, 6595, 6596, 6597, 6598, 6599, 6600, 6605, 6607, 6608, 6609, 6622, 6623, 6625, 6626, 6627, 6747, 7166, 7190, 7194, 7196 (old = 5240)	N/A	N/A	N/A	9.32	not present	5240 (old) present but unrelated	6622 present but unrelated	no number	no number
02.11.0042	The system shall be configurable to accept nightly updates of the annual budget allocations from the DCF Budget Ledger System.	5219, 8744, 8745, 8746, 8754, 8756, 8757, 8758, 8760, 8761, 8763, 8764, 8765, 8766, 8767, 8768	N/A	N/A	9.27	Not present	not present	not present	not present	no number	no number
02.12.0074	The system shall provide the capability to accept ACHA data to determine a client's funding eligibility (e.g., Medicare, Medicaid, managed care plans, and third party insurance).	4137, 6933, 6935, 6936, 6937, 6938, 6939, 6940, 6941, 6942, 6943, 7753 (old = 5237)	N/A	N/A	N/A	Not present	9.29	5237 present but unrelated	not present	no number	no number

RTVM - full	Requirement (The system will:)	Doc Ref - Test Plan Reference Location / Test Case Identifier	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Test case number - map to UAT Results and Integration and system test results (excel) - column A	Test case number - map to UAT Results and Integration and system test results (excel) - column B	Column D - UAT results (excel)	Column E - UAT results (excel)
02.13.0064	The system shall be configurable to track payment categories (e.g., Capitated, Cost Reimbursement, Bundled Rates, Capacity, Utilization) as defined in the FASAMS Data Model Document (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	5198, 8423, 8424, 8427, 8427, 8428, 8430, 8447, 8451, 8452, 8456, 8459, 8472, 8480, 8481, 8484, 8488, 8489, 8490, 8590, 8591, 8593, 8596	N/A	N/A	9.20	Not present	not present	not present	not present	no number	no number
03.00.0081	The system shall produce standard reports as defined by DCF (See the Procurement Document Library referenced in ITN Section 1.15 Supporting Documentation).	7430, 7458, 7459, 7460, 7461, 7465, 7466, 7467, 7474, 7475, 7476, 7477, 7175, 7212, 7176, 7213, 7214, 7222, 7224, 7226, 7232, 7365, 7366, 7367, 7368, 7369, 7371, 7372, 7373, 7374, 7375, 7376, 7377, 7378, 7379, 7380, 7381, 7382, 7383, 7384, 7385, 7386, 7387, 7388, 7389, 7390, 7415, 7416, 7419, 7421, 7425, 7427, 7429, 7257, 7261, 7263, 7264,7266, 7267, 7283, 7316, 7319, 7324, 7325, 7288, 7295, 7296, 7297, 7301, 7302, 7307, 7308,	N/A	N/A	N/A	field empty	field empty	7430, 7458, 7459, 7460, 7461, 7465, 7466, 7467, 7474, 7475, 7476, 7477 - DCF Purchased Bed Occupancy Report 7175, 7212, 7176, 7213, 7214, 7222, 7224, 7226, 7232 - Indigent Clients Served 7365, 7366, 7367, 7368, 7369, 7371, 7372, 7373, 7374, 7375, 7376, 7377, 7378, 7379, 7380, 7381, 7382, 7383,	not present	no number	no number

RTVM - full	Requirement (The system will:)	Doc Ref - Test Plan Reference Location / Test Case Identifier	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Test case number - map to UAT Results and Integration and system test results (excel) - column A	Test case number - map to UAT Results and Integration and system test results (excel) - column B	Column D - UAT results (excel)	Column E - UAT results (excel)
		7309, 7337, 7338, 7339, 6913, 6914, 6917, 6918, 7156, 7155, 7154, 7153, 7152, 7151, 7150, 7149, 7146, 6953, 6950, 6945, 6946, 6934, 7340, 7341, 7313, 7314, 7315, 7317, 7320, 7320, 7321, 7322, 7323, 7327, 7328, 7329, 7330, 7332, 7333, 7334, 7335, 6787, 6789, 6875, 6876, 6877, 6879, 6790, 6791, 6817, 6818, 6838, 6839, 6865, 6867, 6868, 6869						7384, 7385 - Operation Beds, Purchase Beds 7386, 7387, 7388, 7389, 7390, 7415, 7416, 7419, 7421, 7425, 7427, 7429 - Operational Bed Occupancy Rate 7257, 7261, 7263, 7264,7266, 7267, 7283, 7316, 7319, 7324, 7325 - Total Licensed Beds The items below were not found: 7288, 7295, 7296, 7297, 7301, 7302, 7307, 7308, 7309, 7337, 7338, 7339, 6913, 6914, 6917, 6918, 7156, 7155, 7154, 7153, 7152, 7151, 7150, 7149, 7146, 6953, 6950, 6945, 6946, 6934, 7340, 7341, 7313, 7314, 7315, 7317, 7320, 7320, 7321, 7322, 7323, 7327, 7328, 7329, 7330, 7332, 7333, 7334, 7335,			

RTVM - full	Requirement (The system will:)	Doc Ref - Test Plan Reference Location / Test Case Identifier	RTVM - Stage 1	RTVM - Stage 2	RTVM - Stage 3	RTVM - Stage 4	RTVM - Stage 5	Test case number - map to UAT Results and Integration and system test results (excel) - column A	Test case number - map to UAT Results and Integration and system test results (excel) - column B	Column D - UAT results (excel)	Column E - UAT results (excel)
								6787, 6789, 6875, 6876, 6877, 6879, 6790, 6791, 6817, 6818, 6838, 6839, 6865, 6867, 6868, 6869			
no number	The system shall have a way to manage the allowed values for fields in FASAMS.	4474 4553, 6658, 6659, 6660, 6713, 6661, 6662, 6636, 6637, 6638, 6642, 6643	9.19	9.15	N/A	Not present	not present	not present	not present	not present	vocabulary tab
no number	For substance abuse providers, the FEIN for ProviderID must exist in the DCF Provider Licensure and Designations System (PLADS).	7616, 8858, 8875, 8877, 8878, 8883	N/A	N/A	9.34	Not present	not present	not present	not present	no number	no number

Appendix E – Map Test Plan to RTVM and Functional Design Document

Test Case Title	Matrix number/description	Functional Design Document
4548 :: 9.1 :: 3,4 :: Login :: User cannot log in if authentication fails	00.00.0012 - The system shall accept a unique PIN for a client from the DCF ACCESS (MCI) system.	9.1.2.3 - When a user enters their username and password and submits to the security server, if the username and password are found and the account has not yet been verified, then they are given a message that indicates that their account has not yet been verified. 9.1.2.4 - When a user enters their username and password and submits to the security server, if the username and password are not found, then they are given a message that indicates that their username and/or password has not been found.
4550 :: 9.2 :: 1,4 :: Home Page :: Provide user a Home Page upon landing - Data Viewer at Customers	00.00.0012 - The system shall accept a unique PIN for a client from the DCF ACCESS (MCI) system. 00.00.0013 - The system shall provide multiple views (e.g., record, data, extract, analysis) to the FASAMS user interface.	9.2.2.1 - When a user successfully logs into the FASAMS security server, and reaches the FASAMS administrative portal home page, they are provided with a home page that contains an appropriate header based on their roles. 9.2.2.4 - Data Viewer users at the customer level (i.e. DCF) are able to access the following functionality: • Job Management The details of available data and functionality are handled in the respective sections for each area.
4318 :: 9.14 :: 14 :: Submitting Entities Workspace :: Admin users can delete submitting entity records	00.01.0014 - The system shall be 508 compliant for ADA accessibility.	9.14.2.14 - The signed in admin user has the ability to delete submitting entities, on the Administration screen, provided that there are no jobs associated to the submitting entity. Specifics on how this will be accomplished will be added once design is finalized.
4478 :: 9.12 :: 21 :: User Workspace :: Enable User Management functionalities - Reset Credentials	00.01.0014 - The system shall be 508 compliant for ADA accessibility. 03.14.0084 - The system shall provide the capability to create, activate, modify, or deactivate a user to multiple roles (see FASAMS Role Mapping in Supporting Documents).	9.12.2.21 - The signed in admin user has the ability to reset credentials for a user within their privileged access on the Administration screen. Specifics on how this will be accomplished will be added once design is finalized.
4318 :: 9.14 :: NA :: Groups List :: Deleting a Submitting Entity removes the corresponding Group	00.01.0014 - The system shall be 508 compliant for ADA accessibility.	not present

Test Case Title	Matrix number/description	Functional Design Document
5140 :: 9.13 :: 9-11 :: Role Workspace :: Admin users can review, edit and update role profile	00.01.0014 - The system shall be 508 compliant for ADA accessibility. 00.05.0076 - The system shall restrict access to functionality within the system based on configurable FASAMS roles (see FASAMS Role Mapping in Supporting Documents). 03.14.0083 - The system shall provide the capability to define standard user profiles from which individual user roles may inherit privileges.	9.13.2.9 - The signed in admin user has the ability to view the profile of an existing role on the Administration screen, taking them to the role workspace to view the role details or to make necessary changes. Specifics on how this will be accomplished will be added once design is finalized. 9.13.2.10 - The detailed role workspace displays all of the detailed information about each role, organized into logical sections, including sections for the role profile, inherited roles, and associated permissions. The sections provide summary information about each section and the user has the ability to edit the data in any given section. 9.13.2.11 - Editing the role profile section allows editing the role name, type, and description. The role type cannot be changed. In order to utilize a different role type, a new role would have to be created.
6622 :: 9.15 :: 8 :: Vocabulary Versions :: Vocabulary code systems mapping	00.01.0014 - The system shall be 508 compliant for ADA accessibility. No number - The system shall have a way to manage the allowed values for fields in FASAMS.	9.15.2.8 - Within a code system version, the user has the ability to map codes from one code system to another on the Configuration screen. Specifics on how this will be accomplished will be added once design is finalized. Note: An example here would be to add a mapping of a DCF system code to a TEDS code system code to support the TEDS extract. Another example would be to add a mapping of a submitting entity code to a DCF code to support the submitting entity not wanting to change their coding system to match the DCF coding system.
8213 :: 9.16 :: Rules :: Verify 'Director Prefix Name' rule is enforced correctly when enabled	00.01.0014 - The system shall be 508 compliant for ADA accessibility. 00.01.0021 - The system shall allow data validation to be defined using configuration tables (as opposed to hard-coded logic) within the application. No number - The FASAMS system will allow administrators to manage the business rules associated with a particular data set.	Rules are configurable

Test Case Title	Matrix number/description	Functional Design Document
	00.00.0011 - The system shall provide the capability to configure uniform business rules to drive functionality and activities.	
7242 :: 9.11 :: 5-7 :: Jobs List :: Users can sort, filter, and view submitted jobs in detail	00.00.0010 - The system shall provide error codes and code descriptions explaining the reason a submitted record is being rejected (e.g., invalid OCA code, invalid admission date). 00.01.0014 - The system shall be 508 compliant for ADA accessibility.	9.11.2.5 - Each job in the list displays basic information about the job, including Name, Submitter, Date, Status, Successful (# of successful records), and Errors (# of error records). 9.11.2.6 - The job list is sortable and filterable based on the attributes displayed. 9.11.2.7 - The user has the ability to view the job details of any given job on the Job Workspace screen.
5623 :: 9.3 :: 2 :: Submission :: Users can upload one or multiple files	No number - The system shall allow file submissions via the administrative portal .	9.3.2.2 - When a user selects a submitting entity, they have the ability to upload one or more files by browsing their computer's file system.
4611 :: 9.24 :: 7 :: SFTP Upload :: Files contains incorrect entries will cause an upload failure	No number - The system shall allow each submitting entity to utilize an SFTP folder. 00.05.0077 - The system shall provide the ability to perform secure file transfers using a file transfer method (e.g., SFTP, FTPS, SSH).	9.24.2.7 - When the user submits their file(s), any files that are not successfully uploaded are placed into the submitting entity's error file location and a corresponding job is created for the submitting entity with a completed status and description of upload failure. See Functional Area – Submitting Entity Job Processing.

Test Case Title	Matrix number/description	Functional Design Document
8087 :: 9.27 :: 6 :: Database :: IDS Budget interface synchronization retrieves records based on certain criteria	00.03.0039 - The system shall interface with the DCF Budget Ledger system. 00.03.0044 - The system shall interface with the Integrated Data Set system. 02.00.0042 - The system shall be configurable to accept nightly updates of the annual budget allocations from the DCF Budget Ledger System. 02.11.0043 - The system shall be configurable to accept data (e.g., Schedule of Allotment Balance, Schedule of Funds) from IDS. 02.11.0044 - The system shall provide the capability to receive annual budget allocations from the DCF Budget Ledger system. 02.11.0048 - The system shall track budget allocations by Managing Entity. 02.11.0049 - The system shall provide the capability to track OCA fund balances. 02.11.0050 - The system shall track budget allocations by Provider.	9.27.2.17 - could be this one: 1. The IDS budget interface synchronization task will only select records from the view identified in the IDS system for budget information as shown below. No other budget information from IDS will be included in FASAMS. • All budget records for all contracts loaded into FASAMS and available in the budget data, identifiable by contract number are selected. This will handle the budget records for the managing entity contracts. • All budget records for DCF Operated Hospitals are selected based on the Organization Level 2 Code (i.e. IBI). o IBI Code 53 identifies records for Florida State Hospital. o IBI Code 55 identifies records for Northeast Florida State Hospital. o IBI Code 57 identifies records for North Florida Evaluation & Treatment Center. • All budget records for DCF Contracted Hospitals are selected based on the Organization Level 2 Code (i.e. IBI) and identified OCA Code (i.e. OCA) combinations below: o IBI Code 30 and OCA Code LC000 identifies records for West Florida Community Care Center. § These records will be related to Contract AI102 o IBI Code 30 and OCA Code SFET0 identifies records for South Florida Evaluation & Treatment Center § These records will be related to Contract LI807 o IBI Code 30 and OCA Code TCFT0 identifies records for Treasure Coast Forensic Treatment Center. § These records will be related to Contract LI808 o IBI Code 30 and OCA Code 89Z00 identifies records for South Florida State Hospital. § These records will be related to Contract LI809 • The associations shown above for the DCF Operated and Contracted Hospitals are configurable in a back-end database table. There is no user interface to edit that information, however, as discussed, that data could be manipulated without changing the application code.

Test Case Title	Matrix number/description	Functional Design Document
8083 :: 9.28 :: 12,13,14 :: Database : IDS Contract - Original Amount field, Amended End Date field and Amended Amount field	00.03.0040 - The system shall interface with the DCF Contract system. 02.10.0054 - The system shall use the Provider FEIN number on the Contract Record as a unique Provider ID.	9.28.2.12 - Each contract obtained from IDS contains the value of the "Original Amount" field. This field comes from the ORIGINAL_CONTRACT_AMOUNT field in the view identified in the IDS system for FASAMS to use for contract information. This value represents the original dollar amount of the contract between DCF and the contracting entity (e.g. Managing Entity). Note: This value is not expected to change in the source system (CARS), IDS, or in FASAMS when contract amendments occur, however, if it is updated in the source system, it will be updated in FASAMS as well. 9.28.2.13 - Each contract obtained from IDS contains the value of the "Amended End Date" field. This field comes from the AMENDED_END_DATE field in the view identified in the IDS system for FASAMS to use for contract information. This value represents the new end date of the contract between DCF and the contracting entity (e.g. Managing Entity) after having been amended through the amendment process. 9.28.2.14 - Each contract obtained from IDS contains the value of the "Amended Amount" field. This field comes from the AMENDED_AMOUNT field in the view identified in the IDS system for FASAMS to use for contract information. This value represents the new dollar amount of the contract between DCF and the contracting entity (e.g. Managing Entity) after having been amended through the amendment process.
7617 :: 9.35 :: 10-24 :: Database :: Confirm the AHCA license information is pulled into the ProviderLicenseInformation table	No number - For mental health providers, the FEIN for ProviderId must exist in the AHCA licensure database. No number - For mental health providers, the number of licensed beds must be reportable from the AHCA licensure database.	9.35.2.10-24 - this section contains informaiton re; AHCA licenses

Test Case Title	Matrix number/description	Functional Design Document
7617 :: 9.34 :: 3 :: Database :: Confirm only the PLADS licenses with an expiration date of 1/1/2017 or greater are pulled into FASAMS	No number - For substance abuse providers, the FEIN for ProviderID must exist in the DCF Provider Licensure and Designations System (PLADS). No number - For substance abuse providers, the number of licensed beds must be reportable from the DCF licensure information system.	9.34.2.3 - Each valid provider site license identifier will be used to retrieve the related license records from the PLADS system, and to store the information in FASAMS. Only licenses set to expire on or after 1/1/2017 will be retrieved from the PLADS system.
7234 :: 6.1.2.2 :: 1 :: Record Selection Criteria	not present	not present
4533 :: 9.5.2 :: 21 :: Submission :: Confirm Errors are grouped properly	not present	9.5.2.21 - When a job completes with errors its status is set appropriately to indicate it had errors.
5119 :: 9.10 :: 2 :: MCI: Verify MCI Interface Triggers On a Regular Schedule	00.03.0036 - The system shall accept a unique PIN for a client from the DCF ACCESS (MCI) system. 00.03.0037 - The system shall interface with the DCF ACCESS (MCI) system. 00.03.0038 - The system shall provide the capability to batch request unique PINs (one PIN per client in the batch) from the DCF ACCESS (MCI) system. 00.03.0046 - The system shall provide the capability to request a unique PIN for a new client from the DCF ACCESS (MCI) system. 01.00.0001 - The system shall be configurable to use FASAMS demographic records to request and receive a unique identifier and client eligibility data from ACCESS Florida system . 01.00.0005 - The system shall provide the capability to merge duplicate records to a single record with a unique identifier.	9.10.2.2 - The MCI interface synchronization is also triggered on a regularly scheduled basis, no less than once per day.

Test Case Title	Matrix number/description	Functional Design Document
6734 :: 9.25 :: :: Reports :: Submission Summary Report - filter result by beginning/ending created date	00.06.0066 - The system shall have the capability to track the statuses of erroneous records that have been rejected (e.g., rejected and corrected, rejected and deleted, rejected and pending corrections). 00.03.0045 - The system shall provide the capability to report on interface transmissions (e.g., total number of records loaded, date of interface transmission, amount of time to execute the interface transmission, errors, and failures).	not present
5254/4182:: 3.3 :: Reports :: Confirm the users are only able to view data for their assigned groups	not present	not present