**A close up of a logo

Description automatically generated**

**AGILITY TESTER GUIDE**

October 5, 2021

Table of Contents

[Introduction 4](#_Toc83903580)

[System Description 4](#_Toc83903581)

[Key Features 4](#_Toc83903582)

[About This Guide 4](#_Toc83903583)

[Signing In 5](#_Toc83903584)

[Password Setup 5](#_Toc83903585)

[Password Recovery 6](#_Toc83903586)

[Logging In 9](#_Toc83903587)

[Landing Page 10](#_Toc83903588)

[Access the Landing Page 11](#_Toc83903589)

[Passed vs. Failed Pie Chart 12](#_Toc83903590)

[Customize the lookback period: Passed vs. Failed tests data 12](#_Toc83903591)

[Parsing Information 13](#_Toc83903592)

[Customize the lookback period: Parsing Information 14](#_Toc83903593)

[Tests Awaiting Review 15](#_Toc83903594)

[Review Test Results 15](#_Toc83903595)

[Weekly Top 10 Root Causes 16](#_Toc83903596)

[Customize the scope of the Weekly Top Ten Root Causes list 16](#_Toc83903597)

[Tests Trend 16](#_Toc83903598)

[View Tests Trend Details 17](#_Toc83903599)

[Customize the Tests Trend Graph 17](#_Toc83903600)

[Data Integration Requirements 18](#_Toc83903601)

[Requirements for Import Data 18](#_Toc83903602)

[Prediction Source Data: PCAP/PCAPng 19](#_Toc83903603)

[Filtering Requirements 19](#_Toc83903604)

[Format a single PCAP/PCAPng file 19](#_Toc83903605)

[Format multiple files 20](#_Toc83903606)

[Format multiple directories in an archive 20](#_Toc83903607)

[Training Source Data 21](#_Toc83903608)

[Submit training source data 21](#_Toc83903609)

[Encrypted Data: Processing and Secure Storage 23](#_Toc83903610)

[Size of the Training Dataset 24](#_Toc83903611)

[Out-Of-Band Data Types & Submission 25](#_Toc83903612)

[Automatic Ingestion & Predictions 25](#_Toc83903613)

[Automate File Uploads 25](#_Toc83903614)

[Cloud Deployment: GCS Bucket 25](#_Toc83903615)

[On-Site Deployment: SFTP/SCPServer 26](#_Toc83903616)

[Automate Predictions in public clouds - GCS Buckets 26](#_Toc83903617)

[Atomate Predictions on-site - SFTP/SCP server 26](#_Toc83903618)

[Manual Ingestion & Predictions 27](#_Toc83903619)

[Run Manal Predictions via the Agility UI 27](#_Toc83903620)

[Sessions Monitoring 35](#_Toc83903621)

[Change the Session Owner 36](#_Toc83903622)

[Sessions Page Features Overview 37](#_Toc83903623)

[Notifications 39](#_Toc83903624)

[Review notifications 40](#_Toc83903625)

[Delete a notification 40](#_Toc83903626)

[Mark a notification as read 41](#_Toc83903627)

[Mark a notification as unread 42](#_Toc83903628)

[Prediction Results 43](#_Toc83903629)

[Results Overview 43](#_Toc83903630)

[View Prediction Results 44](#_Toc83903631)

[Select test Sessions for review 44](#_Toc83903632)

[View prediction results 46](#_Toc83903633)

[View ML Model accuracy and extractions for a root cause 47](#_Toc83903634)

[Submit ML Prediction Feedback 48](#_Toc83903635)

[Feedback option 1: provide Confidence Level feedback 48](#_Toc83903636)

[Feedback option 2: Challenge the Root Cause 49](#_Toc83903637)

[Send Finalized Feedback 50](#_Toc83903638)

[How to Analyse Predictions 50](#_Toc83903639)

[CCNR or CCBS 50](#_Toc83903640)

[Success Flow of CCNR 50](#_Toc83903641)

[Failed Flow of CCNR 51](#_Toc83903642)

[WRONG PIN 53](#_Toc83903643)

[UE CANCEL 53](#_Toc83903644)

[sip491-P-CSCF-RequestPending 54](#_Toc83903645)

[sip480-MTAS-MisConfig-CBC&CFO 56](#_Toc83903646)

[sip504-P-CSCF-ServerTimeOut 58](#_Toc83903647)

[Labels/classes available on the VoIP model 58](#_Toc83903648)

[Extractions View 61](#_Toc83903649)

[View and filter Extractions 62](#_Toc83903650)

**[Figure 1: AGILITY Prediction Workflow](#_Toc83903369)** [4](#_Toc83903369)

**[Figure 2: The AGILITY Landing Page](#_Toc83903370)** [11](#_Toc83903370)

**[Figure 3: The Passed vs. Failed Report](#_Toc83903371)** [12](#_Toc83903371)

**[Figure 4: The Parsing Information Report](#_Toc83903372)** [14](#_Toc83903372)

**[Figure 5: The Test Awaiting Review Report](#_Toc83903373)** [15](#_Toc83903373)

**[Figure 6: The Weekly Top Ten Root Causes Report](#_Toc83903374)** [16](#_Toc83903374)

**[Figure 7: The Tests Trend Graph](#_Toc83903375)** [17](#_Toc83903375)

**[Figure 8: Archive formatting sample](#_Toc83903376)** [21](#_Toc83903376)

**[Figure 9: Metadata formatting sample (JSON)](#_Toc83903377)** [22](#_Toc83903377)

**[Figure 10: Archive Formatting Sample - Encrypted Data](#_Toc83903378)** [23](#_Toc83903378)

**[Figure 11: Metadata formatting for encrypted metadata submission (JSON)](#_Toc83903379)** [24](#_Toc83903379)

**[Figure 12: Expected File Types Example (ZIP) – GCS Buckets](#_Toc83903380)** [26](#_Toc83903380)

**[Figure 13: Expected Archive Folder Structure – GCS Buckets](#_Toc83903381)** [26](#_Toc83903381)

**[Figure 14: Expected File Types Example (ZIP) – SFTP/SCP Server](#_Toc83903382)** [27](#_Toc83903382)

**[Figure 15: The Notifications Icon](#_Toc83903383)** [40](#_Toc83903383)

**[Figure 16: The Results Page](#_Toc83903384)** [43](#_Toc83903384)

Introduction

AGILITY delivers Automated Root Cause Analysis (**ARCA**). ARCA includes automated packet capture analysis, root cause analysis, and validation. The software includes a Graphical User Interface (GUI) for user interaction and an API (Application Programming Interface) for integrations with external systems. AGILITY is deployable in preproduction and production environments.

System Description

AGILITY is a cloud native product deployable on a public cloud or on-premises. The software provides an intuitive GUI and REST APIs for easy and efficient integration.

The framework is fully containerized and built as a collection of microservices. It is designed to scale horizontally to increase its processing capacity. Its functionality and performance are monitored and alarmed using cloud native tools.

AGILITY loads test results from the network through packet capture (PCAP) logs or text files. Then, using the available Machine Learning Model, the software generates predictions that detail test session results and provide root cause analyses for unsuccessful tests.

**Figure 1: AGILITY Prediction Workflow**

This image depicts the Agility prediction flow. 

AGILITY retrieves data from customer datasets by an automated process. It also supports manual import through the UI.

AGILITY Data Management functionality provides the end user with a quick overview on specific attributes used by the ML model to analyze the call flow.


Key Features

* 3rd party tools integration for federated security and user authentication
* Deployable using Continuous Integration and Continuous Delivery (CI/CD) tools for full traceability and software changes
* Automatic loading of test results from the network
* User-friendly GUI for results analysis
* Failure prediction and root causes

About This Guide

This User Guide explains AGILITY's functionalities. Using this guide, you will be able to:

1. Log in to AGILITY.
2. Upload test results manually or automatically.
3. View results classification using a combination of pre-trained Machine Learning models.
4. Navigate the landing page and manage notifications.
5. Monitor automatic ingestion/prediction sessions.

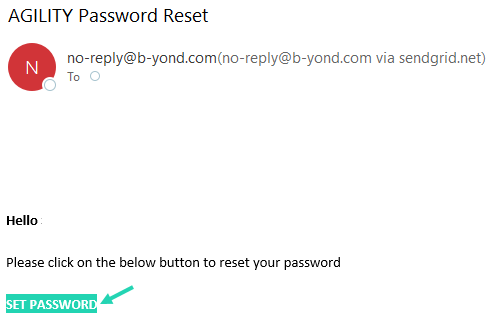
Signing In

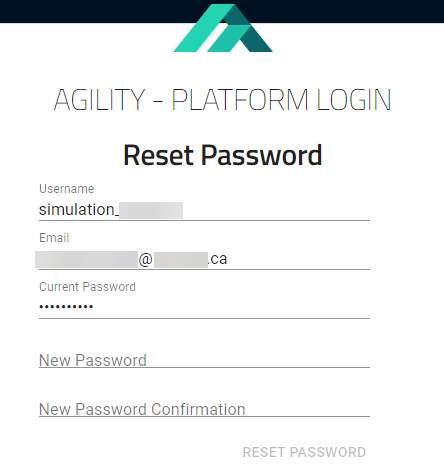
This section describes password setup, password recovery, and how to log in to AGILITY.

Password Setup

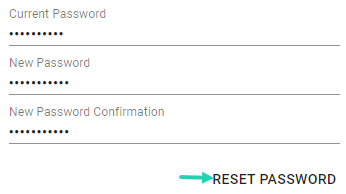
To set up your password:

1. Open your AGILITY account activation email with the subject line **AGILITY Password Reset**.
2. Select **Set password**.



The **AGILITY – Platform Login** page will open in your browser.

1. Type your new password in the **New Password** field and confirm the password by re-entering it in the **New Password Confirmation** field.
2. Select **Reset Password**.



Password Recovery

1. Navigate to the **AGILITY –** **Platform Login** page.



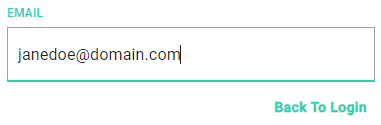
1. Select **Forgot Your Password**.



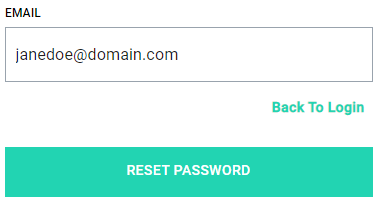
The **Forgot Your Password** page will open.



1. Enter your email address in the **Email** field.



The Reset Password button will enable.

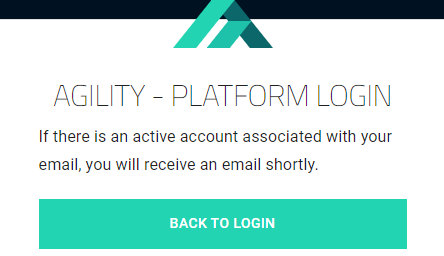


1. Select **Reset Password**.

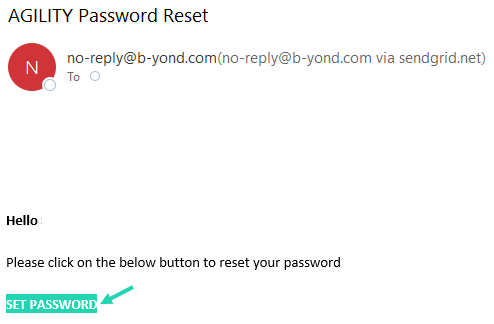
Timeline

Description automatically generated

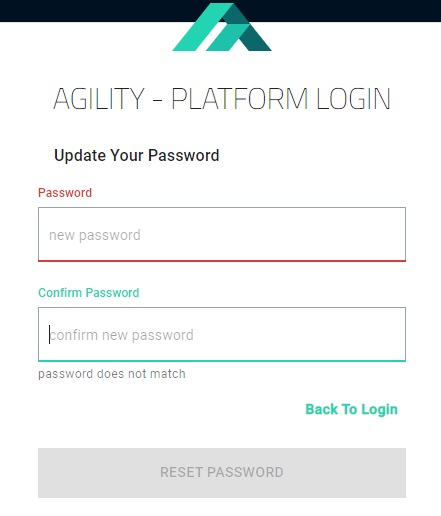
A popup message will appear advising that you will receive a password reset link via email.



1. Go to your email inbox and open your AGILITY password reset email with the subject line AGILITY Password Reset.
2. Select the  link provided in the email.



The Update Your Password page will open.

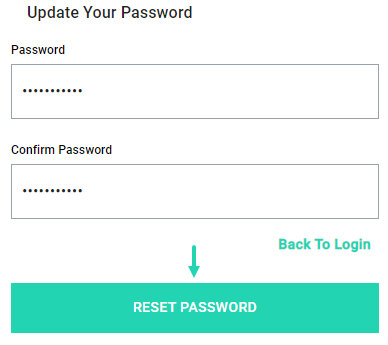


1. Enter your new password in the **Password** and **Confirm Password** fields and verify that both fields match.

**Note**: Passwords must meet the following requirements:

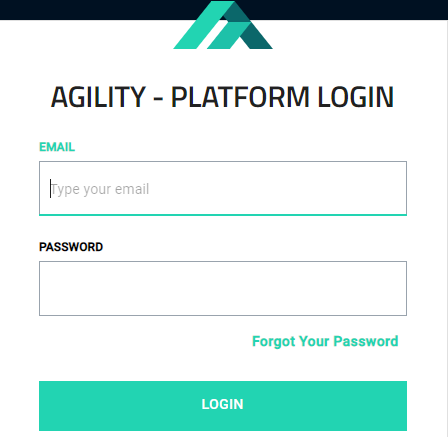
* At least ten characters
* At least one number (ex: 0, 1, 2, 3, etc.)
* At least one lowercase letter (ex: a, b, etc.)
* At least one uppercase letter (ex: A, B, etc.)
* At least one special character (ex: !/”#$%&’()\*+, etc.)

1. Select **Reset Password**.

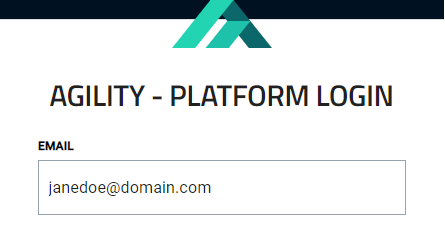


Logging In

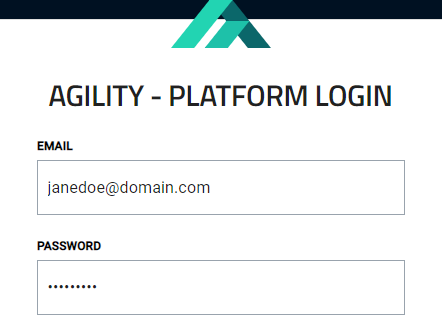
1. Navigate to the Agility – Platform Login page.



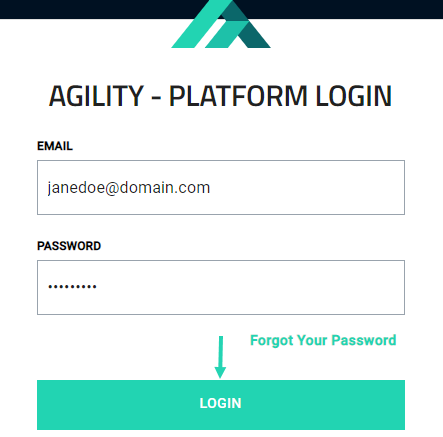
1. Enter your email address.



1. Enter your password.



1. Select **Login**.



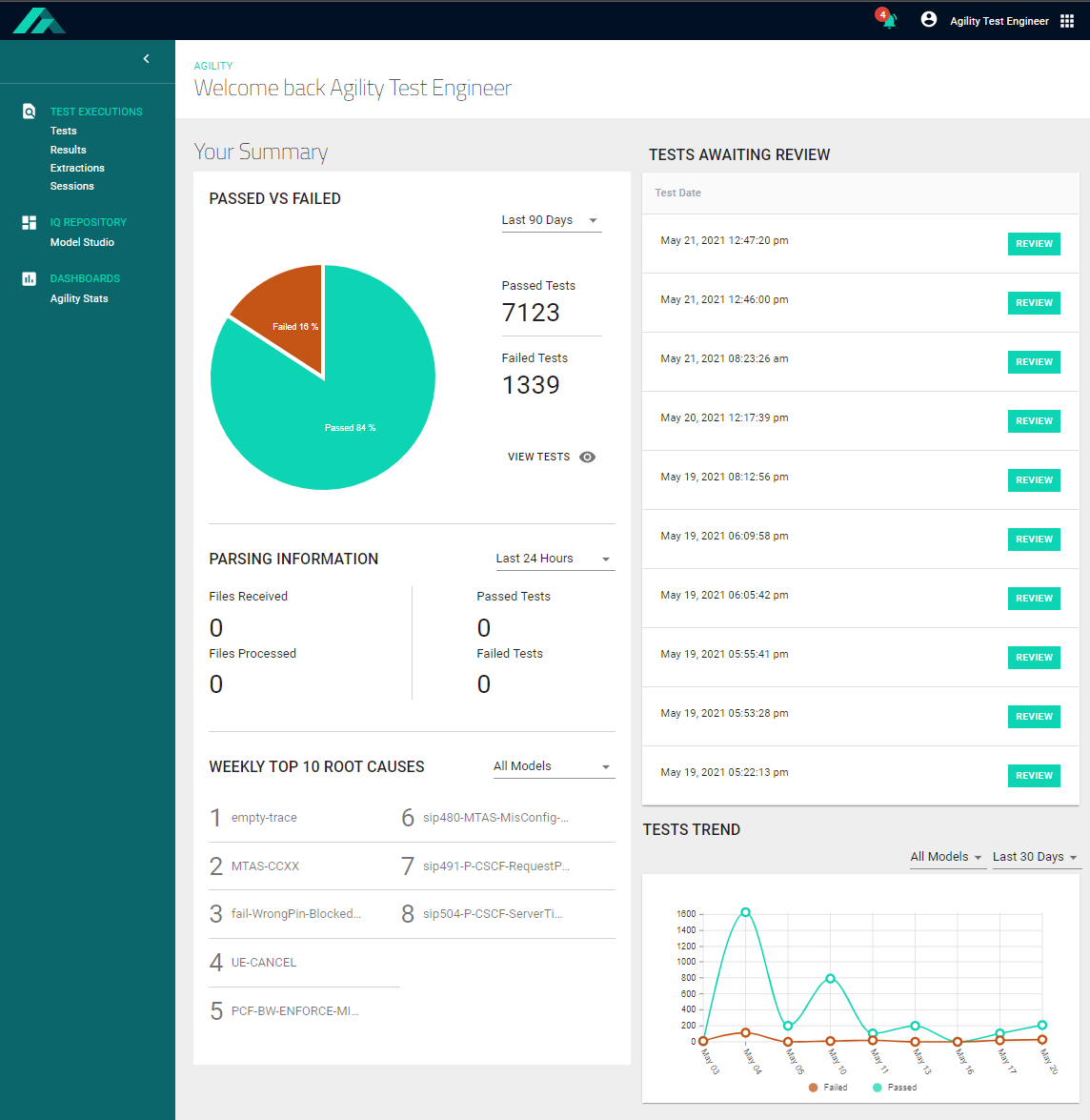
Landing Page

This section outlines the features available on the landing page. The AGILITY landing page provides reports on :

* **Passed VS Failed**: A tally of passed versus failed tests.
* **Parsing Information:** The number of files received and the number of files processed.
* **Weekly Top 10 Root Causes**: The top ten root causes for failures.
* **Tests Awaiting Review:** A list of tests awaiting review. &
* **Tests Trend**: Graph-plotted test trends.

Additionally, the landing page provides users access to the features and actions needed to review Key Performance Indicators (KPIs) and test results.

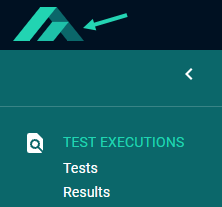
**Figure 2: The AGILITY Landing Page**



Access the Landing Page

To return to the AGILITY landing page from any other location on AGILITY:

1. Select the AGILITY logo  located in the upper right-hand corner of the AGILITY UI.

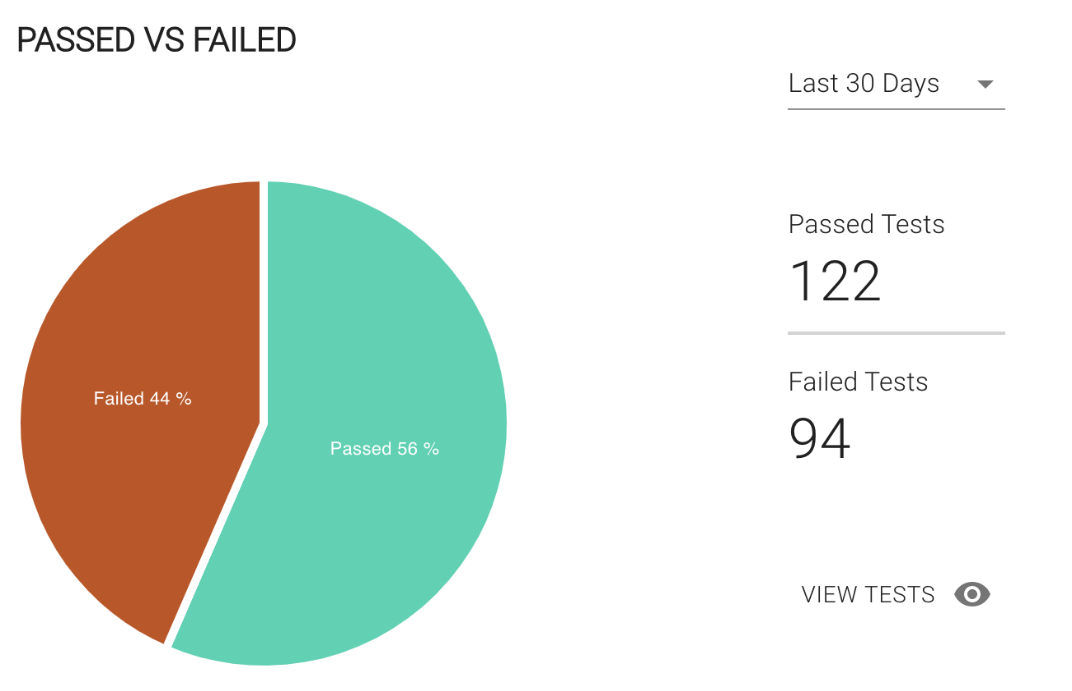


The landing page will open.

Passed vs. Failed Pie Chart

The Passed vs. Failed pie chart provides customizable visualizations of test successes and failures for specified lookback periods. The passed versus failed tests data presented on the landing page includes both automatic and manual session results.

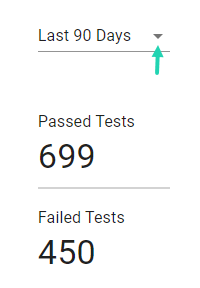
**Figure 3: The Passed vs. Failed Report**



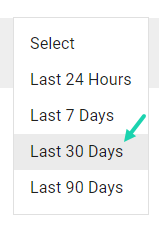
### Customize the lookback period: Passed vs. Failed tests data

To customize the lookback period for passed versus failed tests data:

1. Expand the time frame drop-down.



1. Select the desired lookback period from the list.



The passed vs. failed pie chart and data will update to reflect your selection.

## Parsing Information

The Parsing Information report on the AGILITY landing page provides statistics about:

* **Files Received**: The number of files received by users of the system through manual and automatic processes.
* **Files Processed**: The number of files processed by the system through manual and automatic prediction processes.
* **Passed Tests**: The number of test executions classified as “Passed” by the machine learning model.
* **Failed Tests**: The number of test executions classified as “Failed” by the machine learning model.

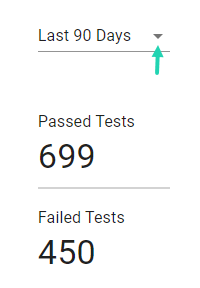
**Figure 4: The Parsing Information Report**



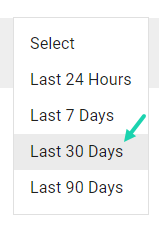
### Customize the lookback period: Parsing Information

To customize the lookback period for the Parsing Information displayed:

1. Expand the time frame drop-down.



1. Select the desired lookback period from the list.

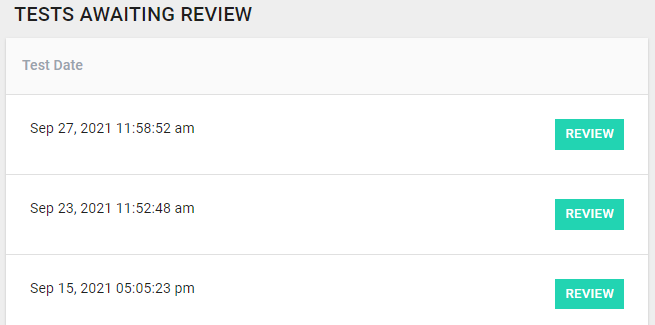


The Parsing Information will update to reflect your selection:

## Tests Awaiting Review

The Tests Awaiting Review report displays the 10 most recent tests pending review. The tests appear in order of descending execution date. Reviewed tests automatically drop-off of the report.

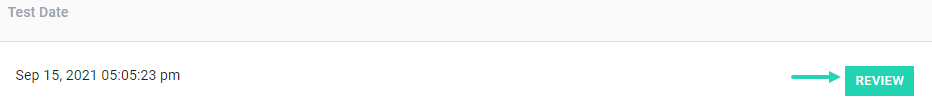
**Figure 5: The Test Awaiting Review Report**



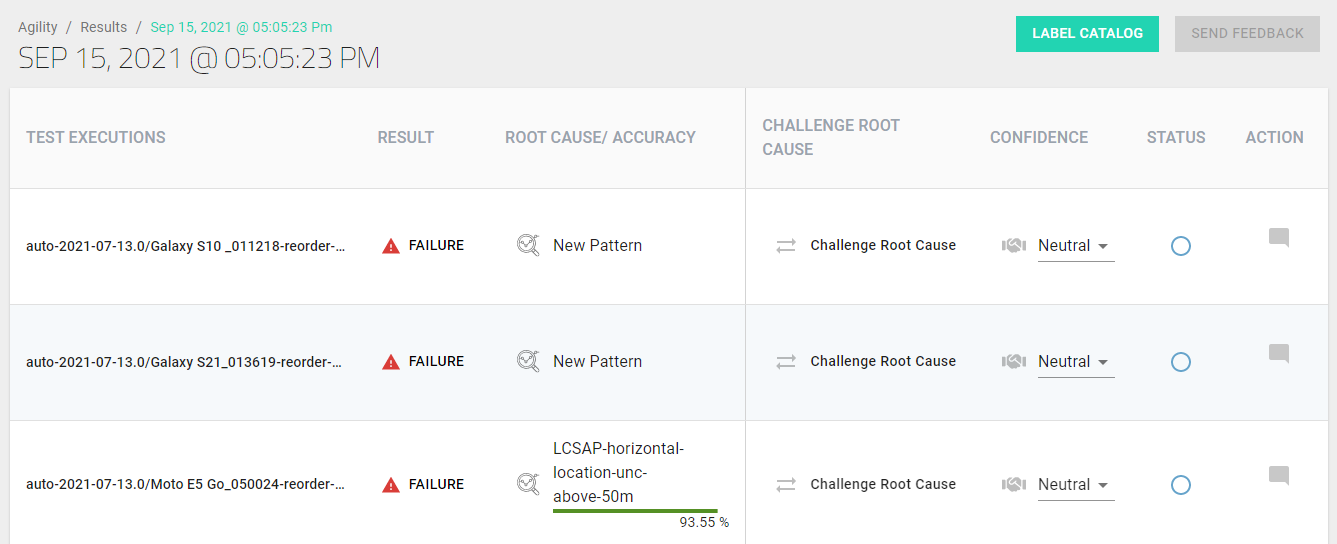
### Review Test Results

To review a test:

1. Select .



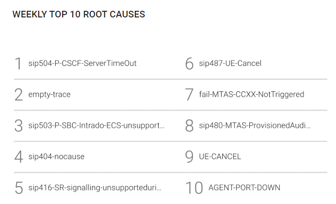
The Results page will open.



## Weekly Top 10 Root Causes

This section provides a customizable list of the Top 10 Root Causes identified by the tests executed during the past week.

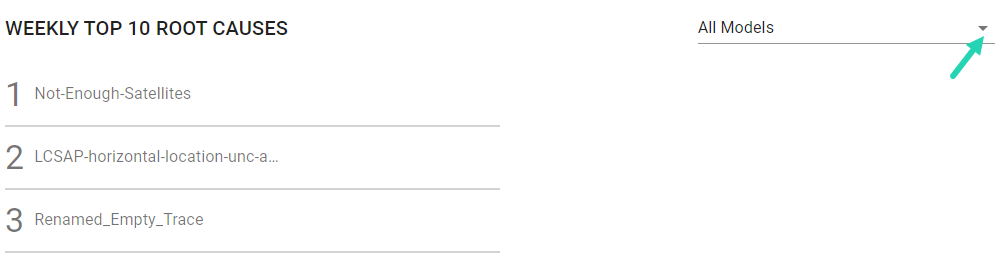
**Figure 6: The Weekly Top Ten Root Causes Report**



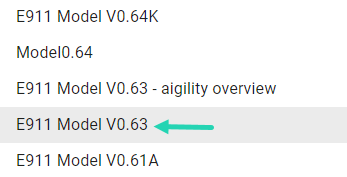
Customize the scope of the Weekly Top Ten Root Causes list

To filter the focus of the Weekly Top Ten Root Causes list:

1. Expand the Model Selection drop-down.



1. Select the desired Model.

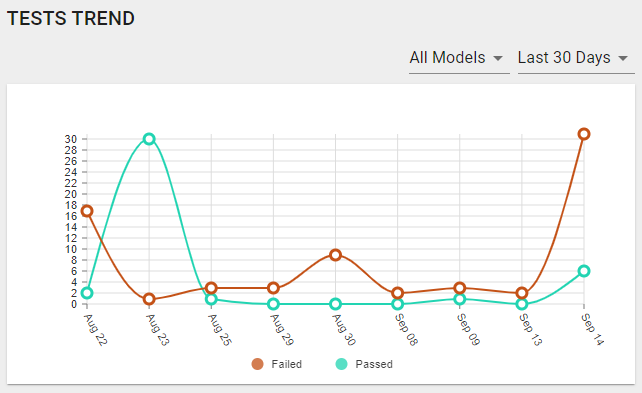


The Weekly Top Ten Root Causes list will update to reflect your selection.

## Tests Trend

The Test Trends plot provides an interactive visualization of Passed vs. Failed tests over time. This feature facilitates efficient failure detection and management. The Tests Trend graph is customizable by Model and lookback period.

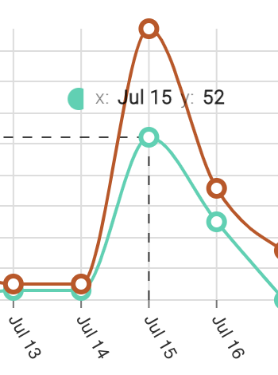
**Figure 7: The Tests Trend Graph**



### View Tests Trend Details

To view details pertaining a data-point on the Tests Trend graph:

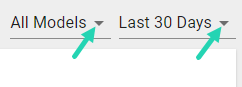
1. Hover over a data point to reveal the corresponding *x* & *y* values.



### Customize the Tests Trend Graph

To customize the ML model and lookback period depicted by the Tests Trend graph:

1. Expand the desired drop-down filter.



1. Select the desired Model/lookback period.

The Tests Trend graph will update to reflect your selection(s).

# Data Integration Requirements

This section describes AGILITY's data format requirements for importing data.

## Requirements for Import Data

AGILITY ingests structured data for prediction purposes (PCAP/PCAPng) and model training (JSON, csv, and Parquet).Table 1 depicts source data format requirements for ingestion.

**Table 1: Data Format Requirements**

| Source File Format | Data Source | Delivery Method | Supported | Description |
| --- | --- | --- | --- | --- |
| PCAP/PCAPng | File System | SFTP API | Y | Packet Capture (PCAP/PCAPng) is a binary file format that captures live network packet data from OSI model Layers 2-7.  Agility accepts one or multiple PCAPs in ZIP archive format. A specific directory structure may be required to accelerate data analysis. |
| JSON | File System | SFTP API | Y | JavaScript Object Notation (JSON) is a standard text-based format used to represent structured data based on JavaScript object syntax.  Agility accepts one or multiple JSON files in ZIP archive format. A specific directory structure may be required to accelerate data analysis. |
| CSV | File System | SFTP API | Only Pre-processing | A Comma-separated values (CSV) file is a delimited text file containing multiple attributes.  CSV files can be used to carry metadata to pre-process the PCAP & JSON file. |
| PSV / Text Column Delimited | File System | SFTP API | Only Pre-processing | Structure Delimited (Pipe or other) text file that can be imported into Agility. |
| Parquet | File System | SFTP API | In progress | Columnar file format for nested data structures. |
| Avro |  |  | Roadmap |  |
| Jflow |  |  | Roadmap |  |
| Netflow |  |  | Roadmap |  |
| Log text files |  |  | Roadmap |  |

## Prediction Source Data: PCAP/PCAPng

**AGILITY** ingests PCAP /PCAPng files for prediction. PCAP /PCAPng files for import can include encapsulated packets starting from the data link layer up to application layer protocols used for prediction purposes.

## Filtering Requirements

Users can filter out unrelated protocol layers, messages, or conversations using any suitable filtering tool such as ***tshark*** before submitting them for ingestion. Similar filtering must be applied across all prediction and training source data.

### Format a single PCAP/PCAPng file

To format a single PCAP/PCAPng file corresponding to a single test for import: f

1. Place the file in a directory.
2. Give the directory a unique name that indicates the corresponding test. For example:

test000-systemundertest/

file-interfaceA.pcapng

1. Submit the directory.

### Format multiple files

Users interested in receiving a prediction for a test involving a collection of multiple PCAP/PCAPng files can submit a directory with all related files included. Users must ensure data integrity, e.g., data in various files are interrelated to the same flow, one or more common identities can be used to trace interesting traffic, etc.

To format multiple PCAP/PCAPng files corresponding to a single test for import:

1. Place all the files related to the test into a directory.
2. Give the directory a unique name that indicates the corresponding tests. For example:

test001-systemundertest/

file-interfaceA.pcapng

file-interfaceB.pcapng

file-interfaceC.pcapng

1. Submit the directory.

### Format multiple directories in an archive

To format multiple directories corresponding to multiple tests:

1. Place the directories into an archive file, e.g., tar, zip. For example:

Archive: testfiles-systemundertest.zip

inflating: test000-systemundertest/

file-interfaceA.pcapng

inflating: test001-systemundertest/

file-interfaceA.pcapng

file-interfaceB.pcapng

file-interfaceC.pcapng

1. Submit the archive.

Training Source Data

ToinformML training,AGILITY ingests directories of raw or filtered PCAP/PCAPng files and relevant metadata.

### Submit training source data

To submit directories of raw or filtered PCAP/PCAPng files along with relevant metadata:

1. Place the PCAP/PCAPng files and metadata into an archive file, e.g., tar, zip.
2. Ensure that the training metadata includes a unique label (that indicates the expected prediction result for the associated set of PCAPng files) as well as any other useful information for filtering interesting parts of the raw data. Below, **Figure 8** depicts an archive formatting sample, and **Figure 9** provides a metadata formatting sample.

**Figure 8: Archive formatting sample**

Archive: trainingset-systemundertest.zip

inflating: trainingset-metadata-systemundertest.json

inflating: training000-systemundertest/

file-interfaceA.pcapng

inflating: training001-systemundertest/

file-interfaceA.pcapng

.....

inflating: training999-systemundertest/

file-interfaceA.pcapng

**Figure 9: Metadata formatting sample (JSON)**

{

"trainingset": {

"trainingData": [

{

"label": "verdictA",

"directory": "training000-systemundertest/",

"trainingFile": [

{

"name": "file-interfaceA.pcapng",

"interesting\_IPAddress": "10.20.30.40",

"startTimestamp": "1579601881",

"endTimestamp": "1579602081"

}

]

},

{

"label": "verdictB",

"directory": "training001-systemundertest/",

"trainingFile": [

{

"name": "file-interfaceA.pcapng",

"interesting\_IPAddress": "10.20.30.40",

"startTimestamp": "1579602567",

"endTimestamp": "1579602678"

}

]

},

....

{

"label": "verdictC",

"directory": "training999-systemundertest/",

"trainingFile": [

{

"name": "file-interfaceA.pcapng",

"interesting\_IPAddress": "10.20.30.40",

"startTimestamp": "1579634567",

"endTimestamp": "1579634678"

}

]

}

]

}

}

Encrypted Data: Processing and Secure Storage

TLS or WPA2 encrypted data can be decrypted (before extraction for

training or prediction processes) through a preprocessing step in AGILITY.

If the encryption key is static, it can be provided out-of-band, i.e., stored securely outside AGILITY and retrieved for deployment.

Dynamic encryption keys (unique encryption keys for each file or group of files) must be provided as a metadata component along with the corresponding file(s) in a structured format. Below, **Figure 10** depicts an archive formatting sample for encrypted data submission, and **Figure 11** provides metadata formatting for encrypted metadata submission (JSON).

**Figure 10: Archive Formatting Sample - Encrypted Data**

Archive: testfiles-systemundertest.zip

inflating: testfiles-metadata-systemundertest.json

inflating: test000-systemundertest/

file-interfaceA.pcapng

inflating: test001-systemundertest/

file-interfaceA.pcapng

file-interfaceB.pcapng

file-interfaceC.pcapng

**Figure 11: Metadata formatting for encrypted metadata submission (JSON)**

{

"predictionset": {

"predictionData": [

{

"directory": "test000-systemundertest/",

"predictionFile": [

{

"name": "file-interfaceA.pcapng",

"encryptionKey": "secret-key01"

}

]

},

{

"directory": "test001-systemundertest/",

"predictionFile": [

{

"name": "file-interfaceA.pcapng",

"encryptionKey": "secret-key02"

},

{

"name": "file-interfaceB.pcapng",

"encryptionKey": ""

},

{

"name": "file-interfaceC.pcapng",

"encryptionKey": ""

}

]

}

]

}

}

Size of the Training Dataset

Like any machine learning system, AGILITY needs training data to generate a model. There is no definitive formula for a threshold for a minimum training dataset. The larger the training dataset, the higher the likelihood of developing a model that makes predictions with higher accuracy. Training dataset size is proportional to the number of classes and the number of features used for classification. Instead of sticking with a specific number, we recommend being prepared to add more training data if more classes are needed for differentiation. Similarly, if data analysts determine that more features need to be factored for model development, the training dataset needs to be expanded.

The quality of the data imported is of equal importance to the quantity of data imported. Specifically, the data must include a diverse set of PCAP/PCAPng files that correspond to a common classification as opposed to multiple copies of the same file.

Out-Of-Band Data Types & Submission

Extraction logic development and model development in AGILITY are performed by data analysts who are subject matter experts in the domain perform extraction logic and model development in AGILITY. Their work is facilitated by the following types of data:

* Host files that provide mapping between IP addresses and hostnames
* Message flow information that depicts success and exception procedures
* Attribute level detail that describes the semantics of the underlying protocol

Such information is not machine processed; it is provided to data analysts in an out-of-band fashion.

# Automatic Ingestion & Predictions

AGILITY offers users an automatic ingestion and prediction pipeline that continuously integrates new test results from the network and generates root cause analyses informed by the most up-to-date data.

This section of the user guide explains how to automate file uploads on-site or on the Cloud.

Automate File Uploads

Automation is AGILITY's guiding principle. Accordingly, AGILITY provides ingestion capabilities to automate the ingestion of data for model training or predictions.

Currently, there are two methods to automate the ingestion of training or prediction input data:

* Google Cloud Storage (GCS) bucket
* SFTP/SCP server

### Cloud Deployment: GCS Bucket

When AGILITY is deployed in public clouds with public Internet access, GCS bucket is the easiest method to upload training and prediction data. In this scenario, a GCS bucket is associated with a service account that allows AGILITY ingestion to monitor bucket content (through directory and filename filters) to automatically read recently uploaded files.

The creation of a GCS bucket service account requires minimal permissions, for example: (storage.buckets.get, storage.buckets.list, storage.objects.get, storage.objects.list.)

GCS buckets can be accessed using [Google's GCS API](https://console.cloud.google.com/) to automate prediction file and training dataset uploads. For more information, please refer to the [Google Cloud Storage API support documentation](https://cloud.google.com/storage/docs/apis).

### On-Site Deployment: SFTP/SCP Server

When AGILITY is deployed on-premises without public Internet access, SFTP/SCP server is the preferred method to upload training and prediction data. In this scenario, a server (physical or virtual) is provisioned with SSHD and a public/private key pair is generated for AGILITY. The only configuration on the SSH server side is to add the public key to the authorized\_keys file and create dedicated directories on the server to place the training datasets and prediction files. On the AGILITY side, the private key is installed and used by the ingestion engine to monitor the sftp/scp server content (per directory and per filename filters) to automatically read recently uploaded files.

The same server can be accessed using SCP or SFTP to automate prediction file and training dataset uploads. For more information, please refer to the [SSH Academy SFTP wiki](https://www.ssh.com/ssh/sftp/).

### Automate Predictions in public clouds - GCS Buckets

1. Drop your files in the related folder.

**Notes:**

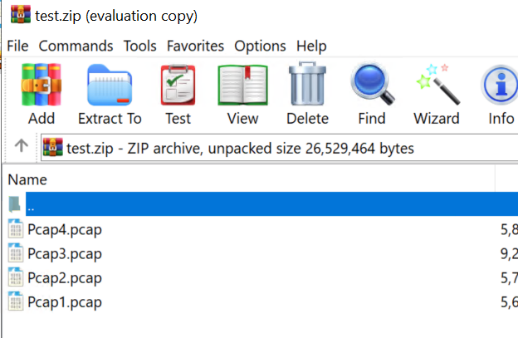
* There must be at least one PCAP/PCAPng file in the folder root.
* Expected file types include ZIP, TGZ, TAR.GZ, PCAP, & PCAPng.

**Figure 12: Expected File Types Example (ZIP) – GCS Buckets**



Expected folder structure inside the archive: test\_execution\_name\_dir → Files (PCAP or PCAPNG):

**Figure 13: Expected Archive Folder Structure – GCS Buckets**



* Depending on the size of your files, it can take up to 30 minutes for the prediction session to appear on the AGILITY Sessions page.

### Atomate Predictions on-site - SFTP/SCP server

1. Drop your files in the related folder.

**Notes:**

* There must be at least one PCAP/PCAPng file in the folder root.
* Expected file types include ZIP, TGZ, TAR.GZ, PCAP, & PCAPng.

**Figure 14: Expected File Types Example (ZIP) – SFTP/SCP Server**

Text

Description automatically generated

* Expected folder structure inside the archive: test\_execution\_name\_dir → Files (PCAP or PCAPNG):

**Expected Archive Folder Structure – GCS Buckets**

Graphical user interface, application, Word

Description automatically generated

* Depending on the size of your files, it can take up to 30 minutes for the prediction session to appear on the AGILITY Sessions page.

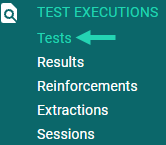
# Manual Ingestion & Predictions

Users can run manual predictions by manually uploading test results through the AGILITY UI. This section of the user guide explains how to run manual predictions using manual file uploads.

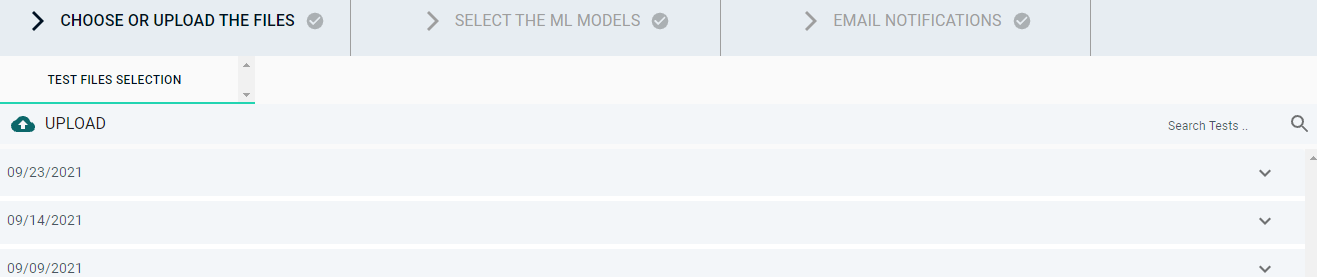
## Run Manal Predictions via the Agility UI

To run manual predictions using manual file uploads through the **AGILITY** UI :

1. Select **Tests** from the side navigation panel.



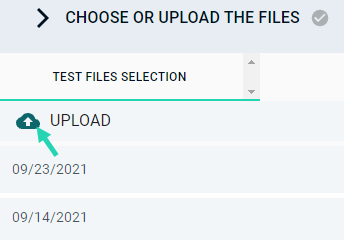
The Test Files Selection page will open.



* 1. Upload new files or folders (i.e., a directory).
  2. Select previously uploaded files.

**Option a):** Upload new files or folders

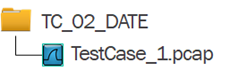
* + 1. Select the upload icon  to upload your file(s) / folder(s) (i.e., a directory)



* + 1. Ensure that your uploads follow the following format:

**main\_test\_name → test\_execution\_*name*\_dir → Files**

* + - * To upload a **single test case file** (for a single call captured in a PCAP or PCAPng file)



* + - * To upload **multiple test case files** (for a single call captured in a PCAP or PCAPng file)



**Note:**

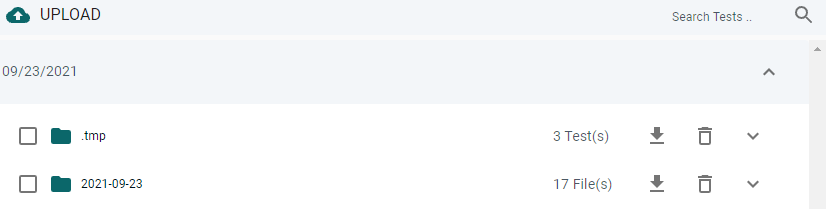
* **You can use any naming convention.**
* **Archives are only supported on the automatic pipeline but not on the manual upload from the UI.**

**Your uploaded files will appear in the list of uploads chronicled by descending (most recent to the oldest) date.**

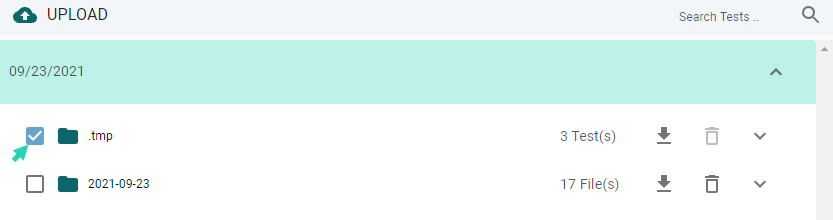
* + 1. **Expand the row that lists the date affiliated with your upload.**



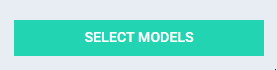
The corresponding uploads will appear.



* + 1. **Select the desired file(s).**



**The option to Select Models for prediction will appear at the bottom of the page.**

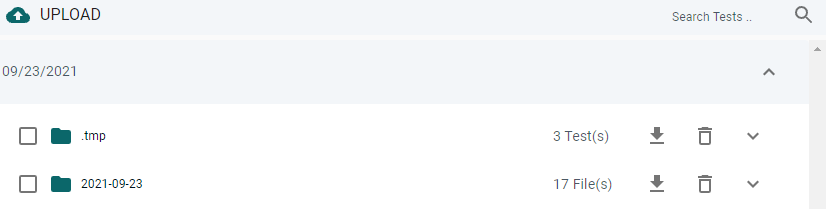


**Option B:** Select previously uploaded files:

* + 1. Locate the desired previously uploaded files
       - Expand the row that lists the date that corresponds to the previously uploaded file.

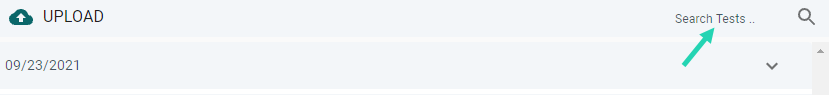


The corresponding uploads will appear.

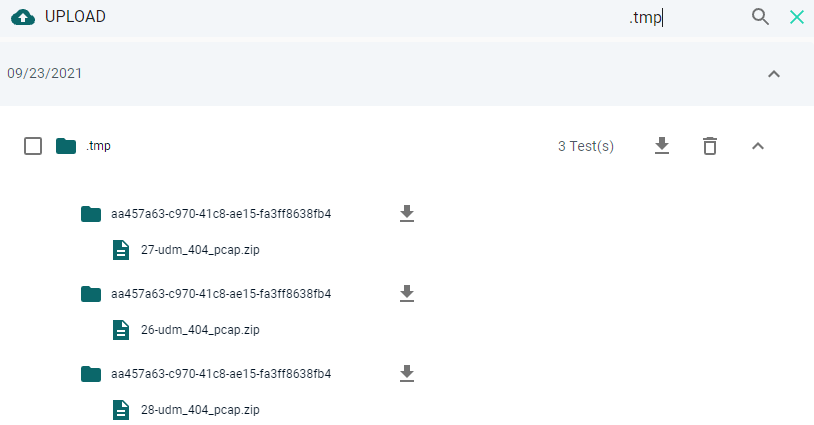


**OR**

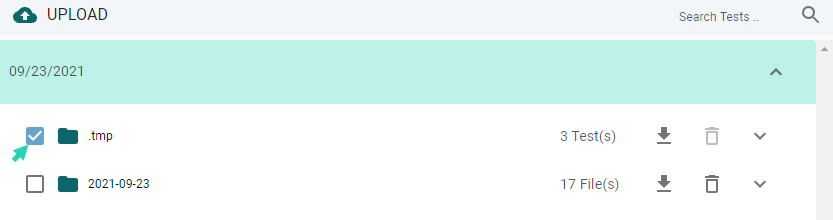
* + - * Use the search feature to query folder names.



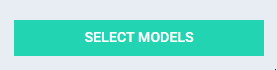
The corresponding uploads will appear.



ii) Select the desired files.



**The option to Select Models for prediction will appear at the bottom of the page.**

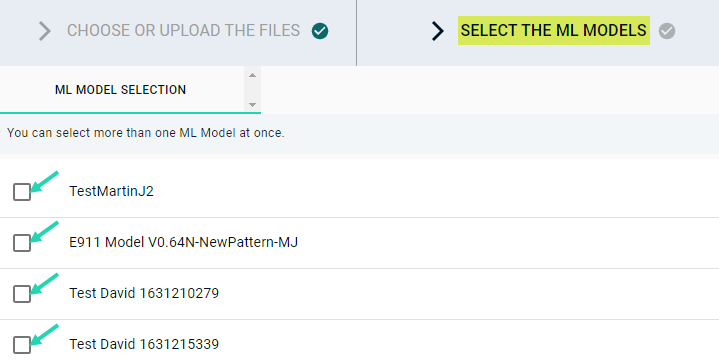


1. Select the **Select Models** button.

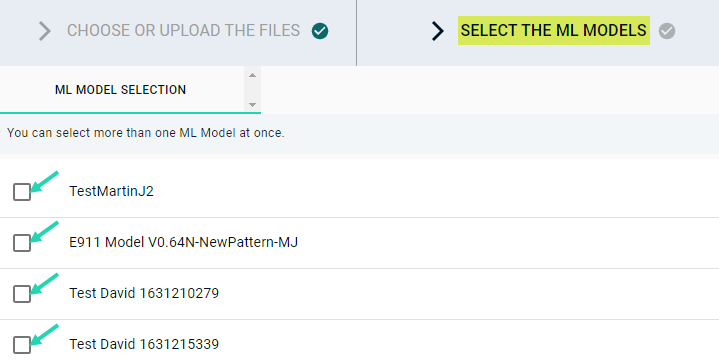
Graphical user interface, website

Description automatically generated

The **Select the ML Models list** will open**.**



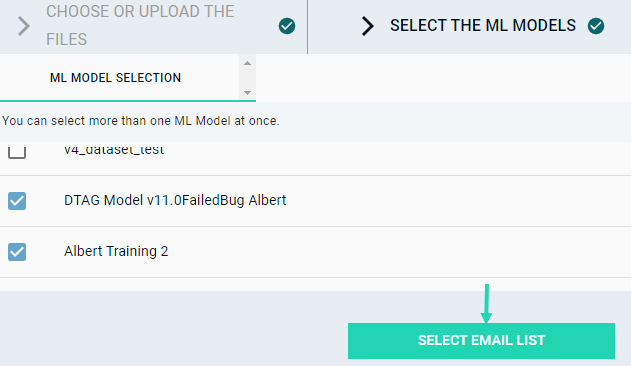
1. Select the desired model or models from the ML Models list. You can select more than one ML Model at once.



The **Select Email List** button will be enabled.

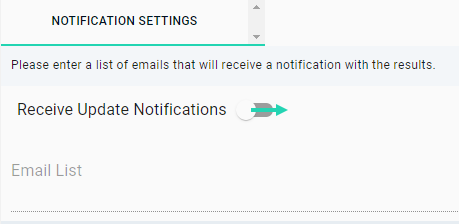


1. Select the **Select Email List** button.



The **Email Notifications** page will open.

1. Input your desired notification settings:
   1. Toggle the **Receive Update Notifications** slider to the right if you would like to receive prediction results notifications.



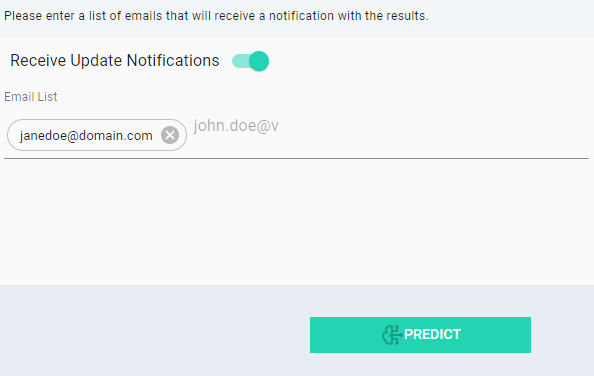
The **Email List** inputfield will be enabled.

* 1. Type in the email address(es) that you would like to be notified when the prediction results are complete. Select **Enter** after inputting each email address.

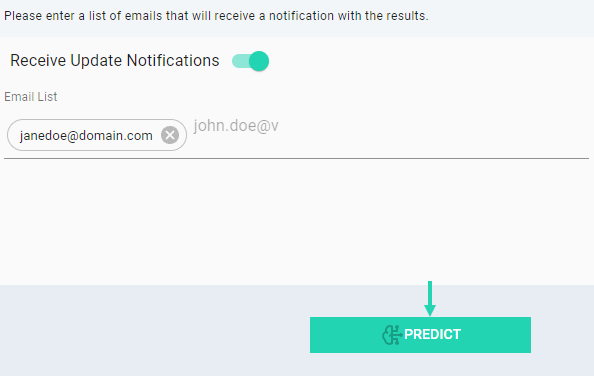
Graphical user interface, text, application, chat or text message

Description automatically generated

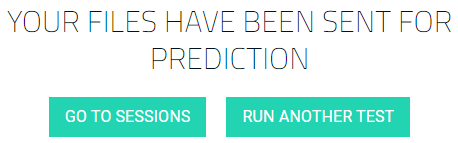
The **Predict** button will be enabled.



1. Select **Predict**.



A prompt message will appear.



1. Select the desired option.  
   1. Select **Go to sessions** to track the status of the prediction process. Once your prediction session is ready, you will receive a notification.
   2. Select **Run another Test** to run another test (using steps 1 – 7 listed above).

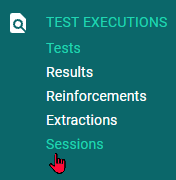
**Note**: If the file size exceeds 20 MB the upload and parsing procedure may exceed 5 minutes.

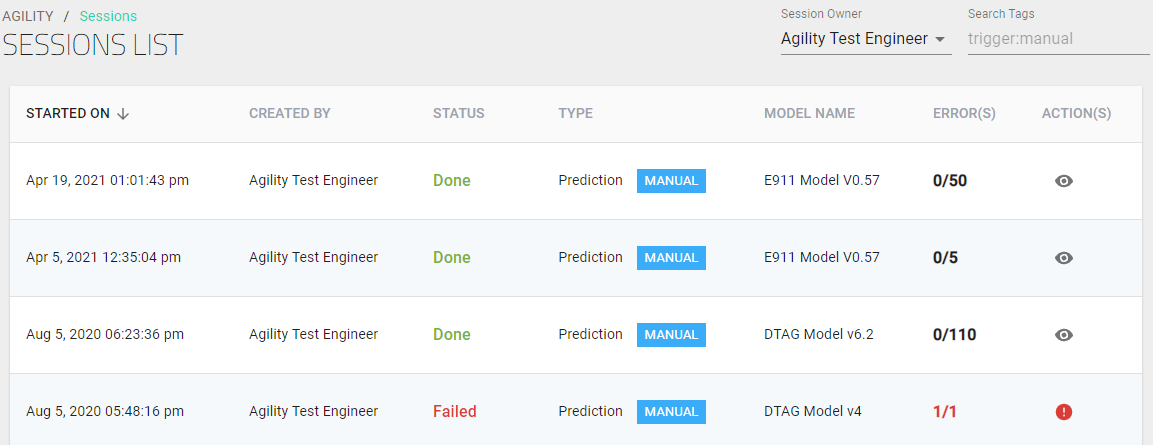
# ****Sessions Monitoring****

The Sessions page displays the current user's Agility Sessions List which provides status and tracking details pertaining parsing and prediction for the sessions run.

To view the Sessions page:

1. Select **Sessions** from the side navigation panel.

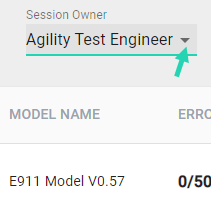


The Sessions page will open and display the Sessions List:

## Change the Session Owner

To view prediction session results categorized under a different Session Owner:

1. Expand the **Session Owner** drop-down.



1. Select the desired Session Owner.

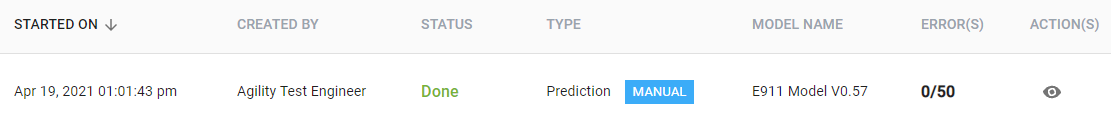
An overview of the selected user's prediction session results will appear.

Graphical user interface

Description automatically generated

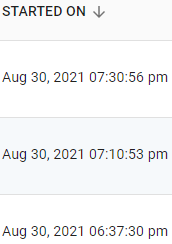
## Sessions Page Features Overview

This section outlines the features available on the Sessions page. The **AGILITY** Sessions page provides a Sessions List overview divided into 7 columns:



The content of each column is explained below.

* **Started On**: The timestamp for initiation of the internal parsing process (does not include preprocessing yet).



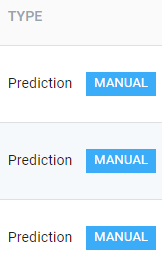
* **Created By:** The name of the user who ran the test. (The default user for automated tests is Agility Test Engineer.)



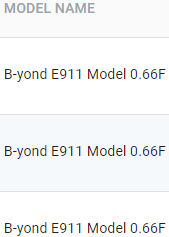
* **Status**:
  + **Started:** The process is still running
  + **Done:** Finished successfully (all files parsed and predicted).
  + **Failed:** No predictions were made for any file, either the parse or prediction process failed.
  + **Warning:** Some files failed to be predicted
  + **Stopped:** The process has stopped

This image depicts the Status column on the Sessions page.

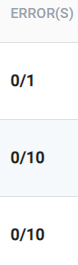
* **Type :** Displays if the session was run manually or automatically and distinguishes between prediction and training session:
  + A **Prediction** session: is a session that uses a machine learning model and a set of test results to generate predictions.
  + A **Training** session: is a session used by subject matter experts to train a model with a training dataset
  + The **Auto** tag  is used to identify a prediction that was executed by the automatic prediction pipeline.
  + The **Manual** tag  represents a manual prediction that was executed by the user directly from the **AGILITY** UI



* **Model Name:** The name assigned to the Model.



* **Error(s)**: A tally of the total errors shown in red font for parsing/prediction failures and in black font for successful predictions.



* **Actions**:
  + Stop predictions. (Available when a test is in progress.)
  + View results (Once a test has finished, the user can select the eye  icon to [view the prediction results](#view_prediction_session_results) on the **Results** page)
  + Download issues (When a Failed or Warning state is present, the user can select the warning  icon to expand the Error Status download page.)



# Notifications

The notifications icon located on the header of the AGILITY UI indicates when Test Status Session updates are pending review. The notifications icon is highlighted with a red a red badge when a user has unread notifications.

**Figure 15: The Notifications Icon**

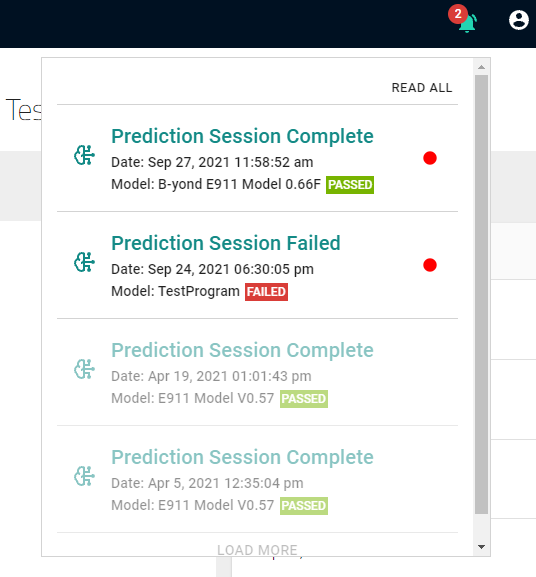


### Review notifications

To review notifications:

1. Select the notifications icon 

The notifications list will open to display the notification’s title, date/time, the Model used for prediction, and the status of each session.

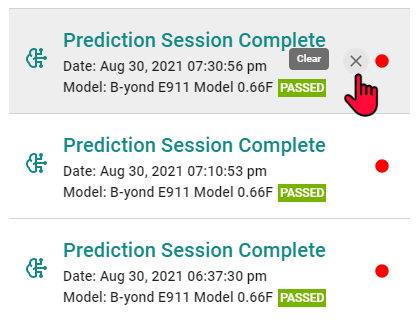


### Delete a notification

To permanently delete a notification:

1. Hover on the left of the red circle (located on the right-hand side of the notification).

The Clear icon will appear:

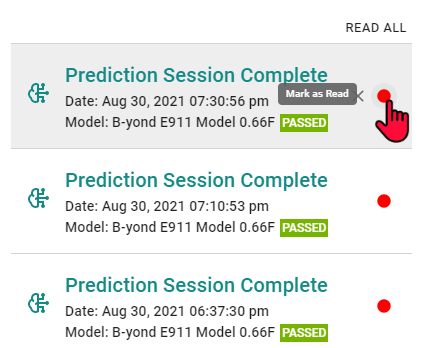


1. Select the 'X'.

AGILITY will permanently delete the notification.

### Mark a notification as read

To mark a notification as read:

1. Select the red circle on the right-hand side of the notification. 

The notification will fade and the circle will turn gray, marking the notification as read. The read/unread indicator will no longer be visible unless the user hovers over the area.

Graphical user interface, text, application, chat or text message

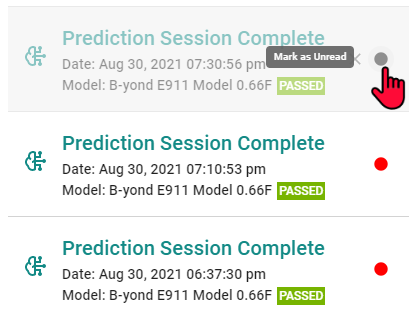
Description automatically generated

### Mark a notification as unread

To mark a notification as unread:

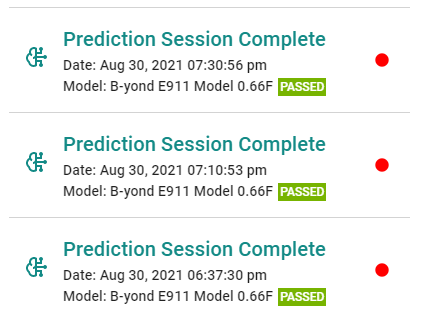
1. Hover on the right-hand side of the notification.

The gray Mark as Unread indicator will appear.



1. Select the gray circle.

The notification will turn red, marking the notification as unread.

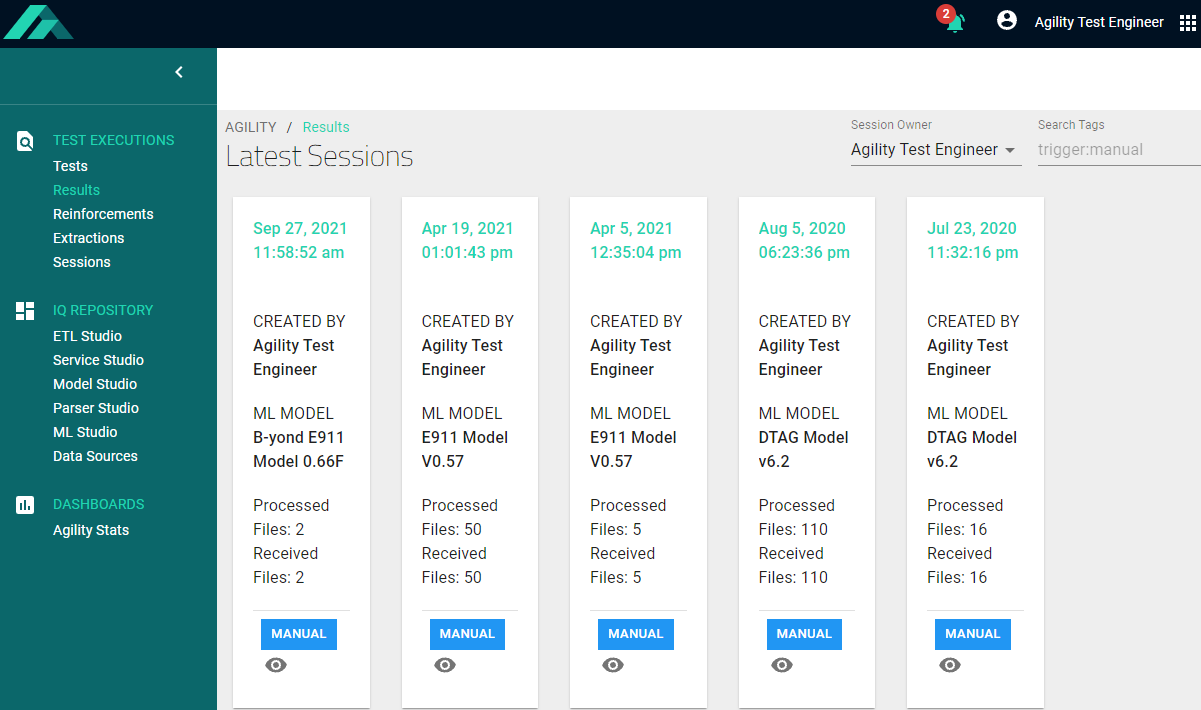


# Prediction Results

## Results Overview

Completed Prediction Session results are accessible through the **Results** page. The Results page provides an overview of the current user's most recent prediction sessions.

**Figure 16: The Results Page**

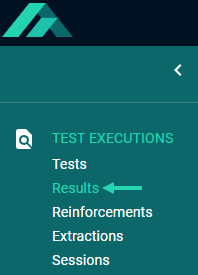


## **View Predict****ion Results**

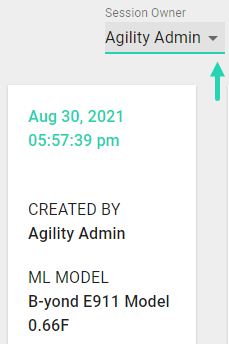
### Select test Sessions for review

To view prediction session results categorized under a different Session Owner:

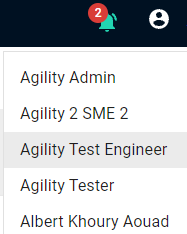
1. Select **Results** from the side navigation panel.



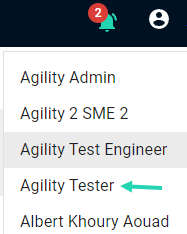
1. Expand the **Session Owner** drop-down.



The list of Session Owners will open.



1. Select the desired Session Owner.

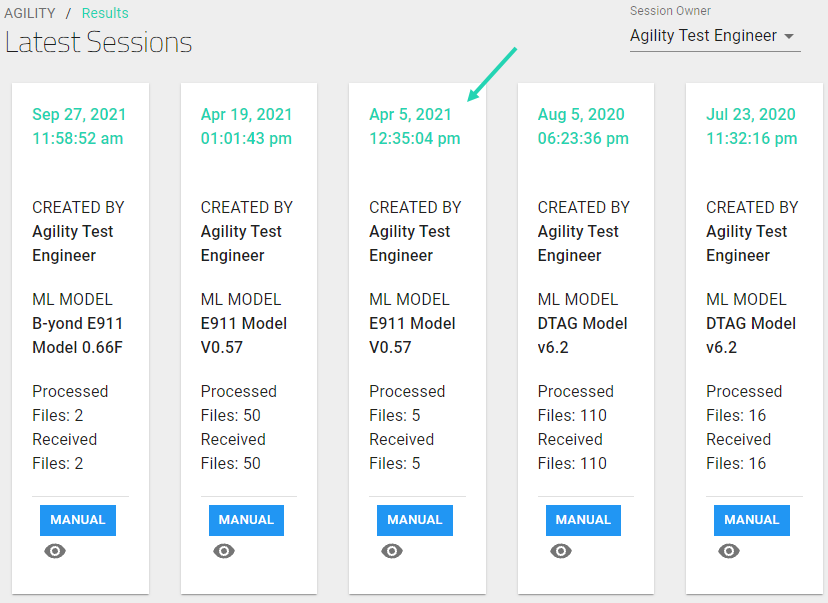


An overview of the selected user's prediction session results will appear.

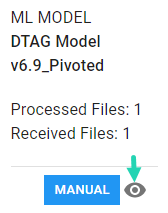
### View prediction results

To view the results for a prediction session:

1. On the Results page, locate the panel that lists the session for review.



1. Select the eye  icon.



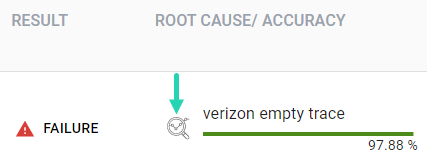
The detailed prediction results for the selected test will open.



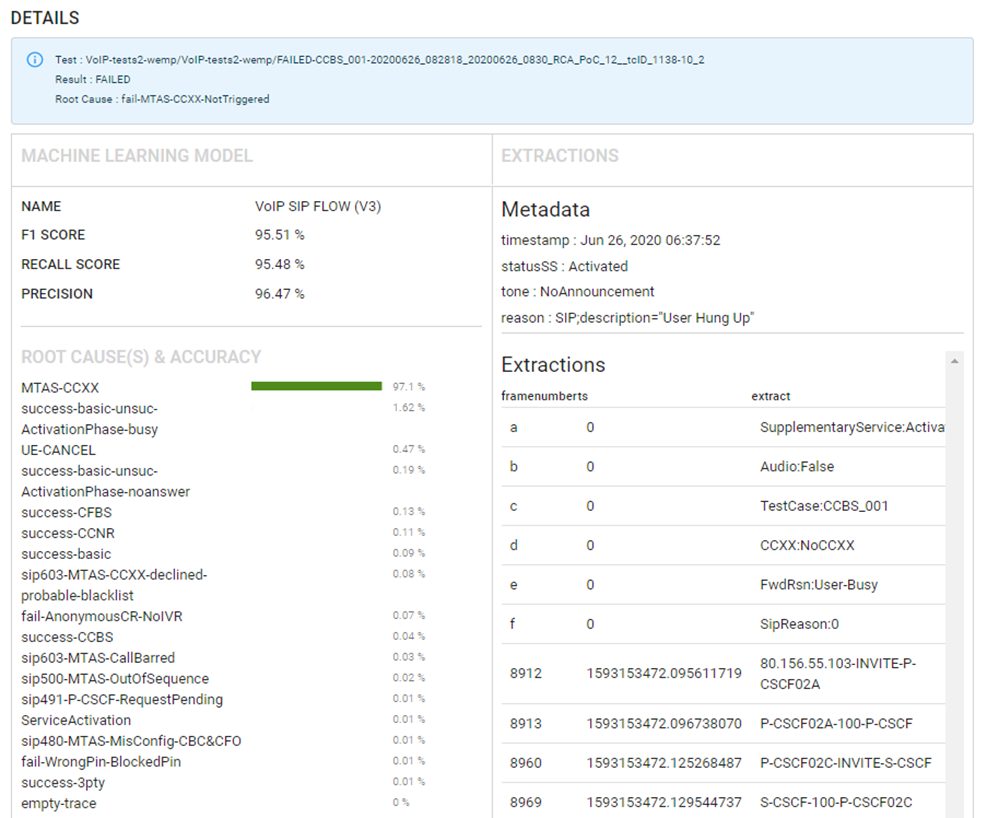
### View ML Model accuracy and extractions for a root cause

To view the ML model accuracy and the extractions for a specific root cause:

1. Select the magnifying glass icon:



The Details page will appear:



## Submit ML Prediction Feedback

Submitting feedback is highly recommended as it contributes to the ongoing development and improved accuracy of predictions. Accordingly, the prediction Results page for each session includes two avenues for providing optional feedback:

* The session prediction **Confidence** drop-down. &
* The **Challenge Root Cause** editor.

After inputting feedback, the user must select the **Send Feedback** button to send the feedback for model reinforcement by B-Yond SMEs.

### Feedback option 1: provide Confidence Level feedback

To provide optional feedback regarding your confidence level in the prediction results for a model:

1. Navigate to the detailed session results page for the Model that you would like to provide feedback for. (Follow the steps listed above in the [Select test sessions for review](#select_test_sessions_for_review) and [View prediction results](#view_prediction_session_results) sections.)

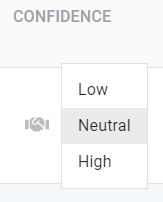


1. Expand the Confidence level drop-down.

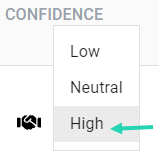
Diagram

Description automatically generated with low confidence

The list of confidence level options will open.



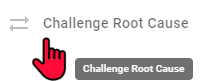
1. Select the confidence level that applies to the Model.



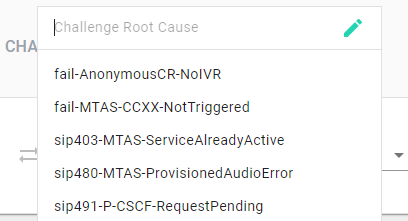
### Feedback option 2: Challenge the Root Cause

To challenge the Root Cause identified:

1. Select Challenge Root Cause.



The Root Cause / Accuracy drop-down will expand.

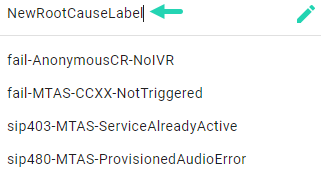


* 1. Select the correct Root Cause from the list.



***OR***

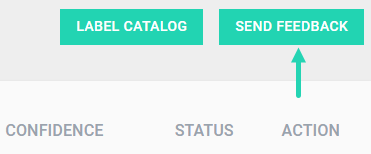
* 1. Create a new root cause label.



### Send Finalized Feedback

To submit finalized feedback after reviewing the results and inputting your assessments:

1. Select **Send Feedback.**



The feedback will be reviewed by B-Yond SMEs and utilized for model reinforcement.

# How to Analyse Predictions

Prediction analysis requires knowledge of the Root Cause labels assigned to test Sessions and an understanding of call log extractions.

This section of the user guide provides:

* An overview of the syntax found in call flow extractions. &
* A Root Cause classifications chart that describes Root Cause labels.

## Call Flow Extraction Syntax

To view the **Extractions** for a test execution, navigate to the Root Cause/Accuracy column of the Results page and select the Icon

Description automatically generated magnifying glass icon to expand the Details page. (For comprehensive procedural steps that delineate how to view Extractions, refer to the [viewing prediction results](#view_prediction_results) section of this user guide.

**Figure 17: Call Flow Extractions**

Graphical user interface, application

Description automatically generated

### CCBS Supplementary Call Service

CCBS (Completion of Calls to Busy Subscriber) is a supplementary call completion feature that appears in call extractions. The (CCBS) service provides outgoing callers with the option to retrigger a call after encountering a busy tone. When the network detects that the callee is unavailable, the caller can activate CCBS by following automated Interactive Voice Response (IVR) prompts.

CCBS automatically informs outgoing callers when the desired callee ends their current call. Once the network detects that the callee is available, the system notifies the caller that the callee is available with a call to the caller. Thus, the CCNR service negates the need for callers to make repeated calls to reach a contact that is on the line with another party on the first call attempt.

The CCBS services call flow is as follows, where **A** is the caller and **B** is the callee:

1. Person **A** calls person **B**.
2. The callee**/B** is on a call. The caller/**A** hears a busy tone.
3. An Interactive Voice Response (IVR) announcement notifies **A** that **B** cannot answer and provides **A** with the option to activate the CCBS Supplementary Service (SS).
4. Once initiated, the CCBS service identifies when the callee/**B** disconnects the call that caused the busy tone active on the network and notifies the caller/**A** that the callee/**B** is available again. If the caller/**A** answers the call, the CCNR service connects the caller/**A** to the callee/**B.**

### CCNR Supplementary Call Service

"Call Completion on No Reply" (CCNR), is another supplementary call connection feature that appears in call extractions. This feature allows a caller to receive an automatic signal when a called party, who could not answer the caller’s first call attempt, resumes activity on the network. When the network detects that the callee is unavailable, the caller can activate CCNR by following automated Interactive Voice Response (IVR) prompts.

CCNR automatically informs outgoing callers when unreachable callees become available again. Network activity identifies when callees become available. Once the network detects that the callee is available, the system notifies the caller that the callee is available with a call to the caller. Thus, the CCNR service negates the need for callers to make repeated calls to reach a contact that is unavailable on the first call attempt.

The CCNR service call flow:

1. Person **A** calls person **B**.
2. **B** does not answer.
3. An Interactive Voice Response (IVR) announcement notifies **A** that **B** cannot answer and provides **A** with the option to activate CCNR Supplementary Service (SS).
4. Once initiated, the CCNR service identifies when the callee/**B** is active on the network and notifies the caller/**A** that the callee/ **B** is available again. If the caller/ **A** answers the call, the CCNR service connects the caller/**A** to the callee/**B.**

### Successful CCNR call flow

A successful CCNR call flow begins with an invite from the callee to the caller.

Graphical user interface, text

Description automatically generated

In this extraction, we see there is an INVITE from A to B, but the call is not being connected. Instead, we see there is an activation of a supplementary service:

**Figure 18: Activation of a Supplementary Service**

Graphical user interface, text

Description automatically generated

After the service is activated, we see the "SUBSCRIBE" of the network to start the monitoring to validate the state for CCNR/CCBS. In this case, CCNR monitoring:

Graphical user interface, text, application, chat or text message

Description automatically generated

There is a "NOTIFY" message from the network that will inform the MTAS the callee is ready to receive a call and the conditions are met, so the MTAS will trigger a new call automatically between A and B with the INVITE with the TriggerCCNR:

Graphical user interface, text, application

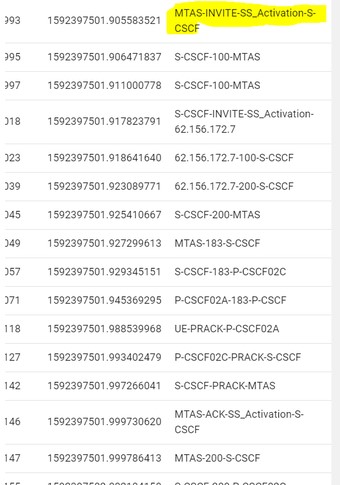
Description automatically generated

## Failed Flow of CCNR

For this extraction, we see there is a call from A to B:



The call is not connected. Instead, A activates a CCNR/CCBS service:



After the service is activated, the call is released:



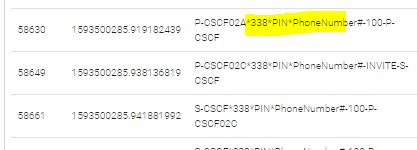
On the flow, there is no """""SUBSCRIBE """""or """""NOTIFY" """" message that belongs to the CCNR/CCBS flow; it means the service fails to trigger and won't work. So, the principal reason that flow failed is that the user has not correctly activated the service, or there is a misconfiguration on MTAS.

## WRONG PIN

This use case is when you have a human error. This is not perceived by the system as it is not an error code but a recorded message on the IVR.

When a user puts in the wrong pin to access a service, a recording is triggered. The reason this use case is important in this scenario is that the first thing checked for was an error on the system, and it can take several hours to focus on a probable human error.

Identify the short number service + PIN the user dials:



Extract the tone to be played for the wrong PIN:



## UE CANCEL

On this extraction, we identify there is a call from A to B:



But before the call is established, A releases the call with a CANCEL:



## sip491-P-CSCF-RequestPending

For this scenario, A calls B:



But the call is being forwarded with a SIP 181 message to C:



Before being connected to the call, there is a new INVITE message, so at that moment user A was in a """""Proceeding" """" state, so after receiving a new INVITE, user A will reject with a SIP491 Request Pending:



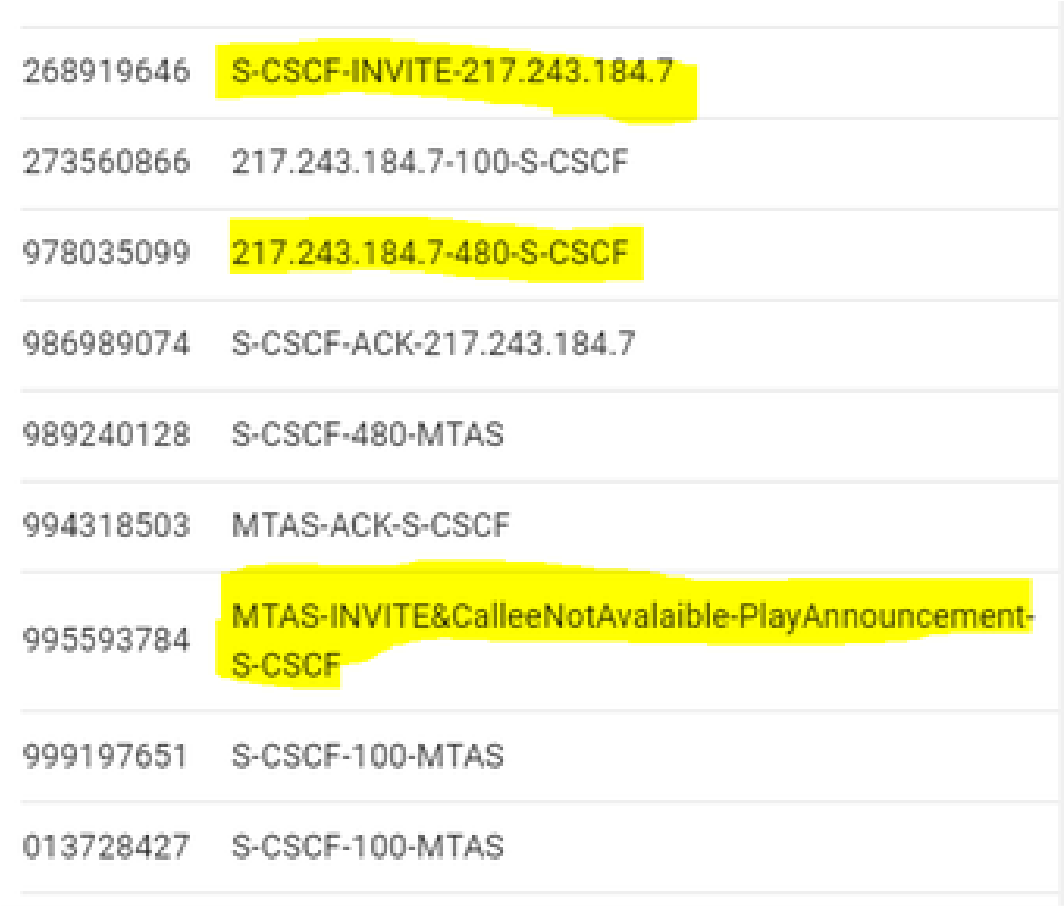
After that error, the call is released. The important thing here is to identify that user A was already trying to establish a call, and during that procedure, it receives a new INVITE.

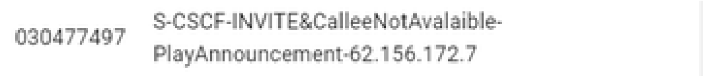
## sip480-MTAS-MisConfig-CBC&CFO

The CBC, Call by Call service, is used to configure a predefined carrier depending on the way the user dials. It is mostly used to have more than 1 carrier to do international calls, so depending on your destination, you can predial a specific number to select the carrier you want to use for the specific international call. In this extraction we see there is a call from A to B:



The call is being sent to the carrier, but the carrier is rejecting the call with a SIP 480 and the MTAS is playing an announcement because the call was rejected:





The reason the call was rejected is that on the MTAS there was a misconfiguration to select the correct international carrier, so the call was sent to the wrong carrier and then rejected with a SIP480 error; that is a human error in the configuration on MTAS.

## sip504-P-CSCF-ServerTimeOut

This scenario is a problem of connectivity. In the extraction, we see there is an INVITE from UE to PCSCF. The next message should be the INVITE being sent from P-CSCF to S-CSCF, but instead there is a SIP 504 error from P-CSCF to UE:



The reason for this error is that the SIP links between the P-CSCF to the other network elements are down, so that could be a physical issue on the interface or an IP problem on the network.

## Labels/classes available on the VoIP model

The following root cause classifications are available in the model:

|  |  |
| --- | --- |
| empty-trace | Label assigned by the model when there is no data to be parsed.  Cause: No interesting data is contained in the trace; thus, nothing was extracted. |
| fail-AnonymousCR-NoIVR | Label used in training with the following scenario:   1. A calls B as anonymous, MTAS does not play an announcement, A releases the call.   Cause: Announcements are mandatory for anonymous calls. The call was rejected due to the absence of the required IVR announcement for anonymous calls.  Instead, the call rings normally and there is no announcement indicating the anonymous call. This error is caused by a misconfiguration on MTAS or a missing activation of the Anonymous Call Rejection on the callee's side. |
| fail-MTAS-CCXX-NotTriggered | Label assigned by the model with the following scenario:   1. A calls B, B is busy or doesn't answer. 2. A activates a CCBS or CCNR service. 3. The conditions are met to trigger the call for CCBS or CCNR. 4. The new call is not generated.   Cause: Wrong activation or misconfiguration on MTAS. |
| fail-WrongPin-BlockedPin | Label assigned by the model with the following scenario:   1. A dials a short number to activate a service, like adding a new number to a black list. For this kind of service, the user will have to dial a unique PIN to activate the supplementary service. 2. The PIN used is incorrect or blocked.   Cause: User is using an incorrect PIN, there is no PIN associated with the user, the pin is blocked. |
| ServiceActivation | Label assigned by the model with the following scenario:   1. A dials a short number to activate a service like CFU, CFNR. 2. The service is activated. |
| sip480-MTAS-MisConfig-CBC&CFO | Label assigned by the model with the following scenario:   1. A calls B with a prefix to select a carrier. 2. The call is being sent to a wrong carrier. 3. The call is rejected from the carrier with a sip480 error.   Cause: Misconfiguration on MTAS to identify the prefix dial to choose the correct carrier. |
| sip491-P-CSCF-RequestPending | Label assigned by the model with the following scenario:   1. A calls B. 2. Before the call is connected, A receives a new INVITE from C. 3. A will reject the call with a SIP 491.   Cause: During a call establishment, the user state is "Proceeding" so it is not able to process a new call. |
| sip500-MTAS-OutOfSequence | Label assigned by the model with the following scenario:   1. A calls B. 2. The call is established, but at some point in the call flow, each message must contain a "SEQUENCE" number; this SEQ number is validated by each network element, and if it doesn't match, the call is rejected with a sip 500 error "Out Of Sequence"   Cause: The main reason of this error is that there is packet loss on the network, so at some point the retransmission messages will increase or modify this SEQ number. |
| sip504-P-CSCF-ServerTimeOut | Label assigned by the model with the following scenario:   1. A calls B. 2. The P-CSCF rejects the INVITE after a few seconds with a SIP 504 error.   Cause: There is a problem on the network; it could be a physical issue on the network port, or an IP problem on the network. |
| sip603-MTAS-CallBarred | Label used in training for the following scenario:   1. A calls B, at some point MTAS rejects the call with 603. Similar to a probable call list, but no audio is played.   Cause: For this case, B should have a Barring List to reject calls from specifics numbers, A must be on that list, so A calls B, the call is rejected but there is a misconfiguration on MTAS to play the correct announcement indicating the call is being rejected by Barred List. |
| sip603-MTAS-CCXX-declined-probable-blacklist | Label used in training for the following scenario: A number is declined due to being on a BL. Cause: For this case, B should have a Barring List to reject calls from specifics numbers; A must be on that list, so A calls B, the call is rejected but there is a misconfiguration on MTAS to play the correct announcement indicating the call is being rejected by Barred List. |
| success-3pty | Label used in training for the following scenarios:   1. Triggered by initiator. 2. Triggered by conference participants. 3. Hold / Toggle status. |
| success-basic | Label used in training for the following scenarios:   1. A calls B, B forward to C, A connects with C, then after some seconds, A releases the call with C. 2. Basic call between A-B. 3. A calls B, B is in Busy State and forwards to C, A connects with C. 4. A calls B, call is connected, A releases call. There is no audio. 5. A calls B, A connects with B, releases the call. 6. A calls B, call is established, A releases call. |
| success-basic-unsuc-ActivationPhase-noanswer | Label used in training when having the following scenarios:   1. A calls B, gets audio 35.wav USER NOT AVAILABLE. 2. A calls B, call is connected, C calls A, tries to connect but A rejects with 480 and then 408 plays tone 35.wav. |
| success-basic-unsuc-ActivationPhase-busy | Label used in training for the following scenarios:   1. A calls B, A gets 486 sip message. No audio.   Cause: A dials to B using a prefix to activate some supplementary service, but the service is not activated, there is no audio played to notify the service activation fail, and the call is released. This is a misconfiguration on MTAS to trigger the correct audio tone. |
| success-CCBS | Label used for unique CCBS scenario:  A calls B, B is busy, A activates CCBS, when B is available again an INVITE is sent with B to connect the call with A. |
| success-CCNR | Label used for unique CCNR scenario. |
| success-CFBS | Label used for unique CCBS scenario:  A calls B, B rejects 486, MTAS sends 181 Forward to C, A is connected to C. |
| success-CWH | Label used in training for the following scenarios:   1. A-B, C calls A, A put on hold B, A talks with C and release call, then A re-establish call with B and release call. 2. A calls B, B rejects the call with 603. We can relabel this one, the one who rejects it is the A and not the callee. 3. A calls B, C calls A, A put on Hold B and connects with C, always plays 68.wav. 4. A calls B, C calls A, A rejects C with 603, there is no audio. |
| UE-CANCEL | Label used in training for the following scenarios:   1. A calls B, the call is ringing, A releases the call before connection. 2. A calls B, B forwards to C, before connecting with C, A releases the call.   Cause: This example is perfect to show the customer a general classification can be generated when there is not enough data to do a deeper analysis. |

# Extractions View

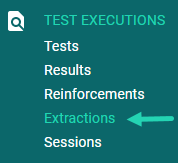
Extraction data is accessible for review through the AGILITY UI and REST APIs. This user guide explains how to view and filter extraction data through the UI. The **Extractions** feature on AGILITY displays the data extracted from imported PCAP files and provides filters for pinpointing a specific subset of the call-flow data for viewing, analysis, or processing. The following filter options are available:

* timestamp
* phone (The phone number of the person who initiated the call)
* useragent (The device and software version used.)
* trunkcontext (A parameter namespace that specifies error codes.)
* extractedreason
* ULPprecsisionmajor
* ULPprecisionminor
* MLP (Mobile Location Protocol)
* label
* framenumber
* ts
* extract

### View and filter Extractions

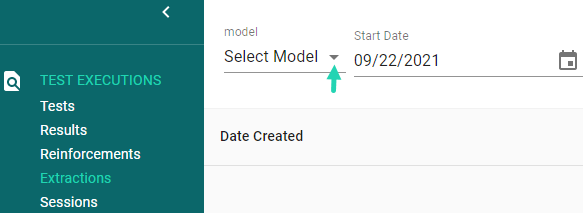
To view & filter Extractions:

1. Select **Extractions** from the left navigation panel.

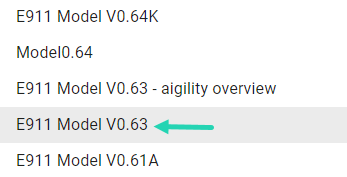


The Extraction page will open.

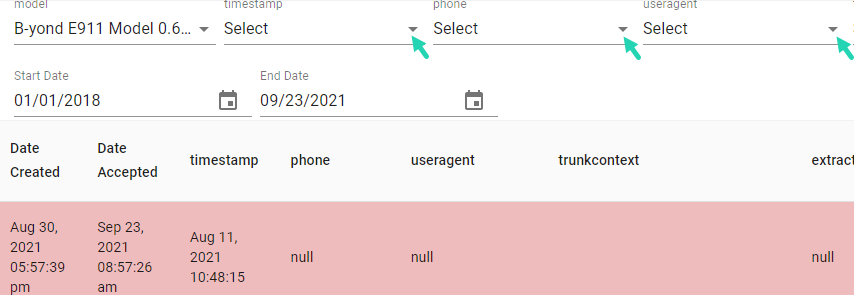
1. Expand the **Select Model** drop-down.



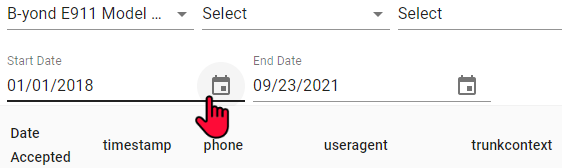
1. Select the desired Model.



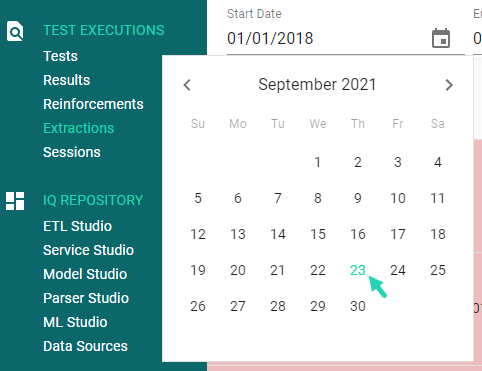
1. Apply filters to narrow your search:



1. Expand the calendar icon for the Start Date.



1. Select the Start Date for the data subset that you would like to review.



The data displayed on the Extractions page will reflect the parameters of your filter and date range selections.