AD SUMMATION
DII/eDII Guide

A Guide for Service Bureaus

Published: June 2004
Updated: June 2011

ABSTRACT

This document provides information about the AD Summation DII/eDII file to service bureaus. The Table of Contents serves as an outline of the electronic discovery (eDiscovery) workflow from the perspective of a service bureau. This document also discusses changes to the structure of the DII file that allow for the batch loading of electronic discovery, and provides a reference of new DII tokens used for email messages and electronic documents. This document assumes prior knowledge of DII files, their structure, and tokens.

This document is geared toward service bureaus that deliver data and eDiscovery to the client in the form of native files, images, full-text, fielded data, or any combination thereof.

Contents

Contents

Contents

Contents

Introduction

AUDIENCE

This document is intended for service bureaus that process electronic discovery and provide DII files to their clients. It assumes a basic knowledge of DII files and their structure.

This document enables service bureaus to more easily provide their AccessData clients with a DII file that will take full advantage of AccessData's new eDiscovery features. A DII file that also loads eDiscovery data is sometimes referred to as an eDII file. This document is designed to assist service bureaus whether they choose to use commercially available software or applications developed in-house to generate a DII/eDIIfile.

STYLES USED IN THIS DOCUMENT

This document provides a number of visual cues to help guide you. The following styles are used:

Italicized Text – Italicized text indicates a term that is specific to DII files or to AD Summation. The first time that a term is used that might be new to you, it is italicized and accompanied by a definition. Italicized text also indicates the title of another document or section within this document.

Bold Text – Bold text indicates an item that is found on the AD Summation interface, such as a menu option, a window, a field, or a dialog box.

`Courier New Font` – Text styled in Courier New font indicates text that you should type as a user and is also used for sample code.

STYLES USED IN THIS DOCUMENT

Note: *Notes call attention to supplemental yet important information about the topics covered in this document. Notes also provide suggestions on how to deal more effectively with electronic discovery. Some suggestions focus on AD Summation's tools, while others have a broader scope.*

SPECIAL CHARACTERISTICS OF EDD FILES

Once information has been created in or converted to an electronic format, it takes on very different characteristics. Electronic information is handled differently, and may contain information of greater value than analog or paper information.

Electronic information is not just text or data, but includes audio, video, and graphics. The sheer volume of electronic information can be staggering. Through routine use and due to the immense storage capabilities of today's computers, thousands or millions of electronic items can be created on a monthly, weekly, or even daily basis.

EMAIL MESSAGES AND EMAIL ATTACHMENTS

Email is the most commonly encountered type of electronic data. It is important to remember that one cycle of data collection may contain several duplicate email messages. Therefore, de-duping by a service bureau can be extremely valuable in reducing the amount of electronic data that a client has to review. In a set of electronic documents, where the volume of email messages can be substantial, the email metadata can often be invaluable for authentication of evidence.

Sometimes email messages include attachments, which are files transmitted or exchanged by being attached to email messages. Email attachments may include forwarded email content in a number of file formats, word processing documents, spreadsheets, and the like. Essentially, attachments are any type of file that the author and recipient need to communicate about or collaborate upon.

Parent/Child Relationships

An email message or other electronic file that has a file attached is referred to as a *compound* document. An email message that has an attachment is called the parent document and the attachment is called the *child document*. It is important to preserve the parent/child relationship of electronic data, which can be reflected in AD Summation document database summaries.

In AD Summation, the separate database summaries created for an email message and its attachment(s) constitute a complete compound document. For example, when a user views an email message in Microsoft Outlook or Lotus Notes, the attachments are an essential part of that email message. In AD Summation, although a compound document is separated into several database summaries – one for each member that is part of the compound document – the entire compound document is considered to be one unit.

Electronic Documents

Note: Specific file types handled as native files in AD Summation are listed later in this document.

Electronic documents are a user's electronic files that are exchanged directly in the investigative and discovery processes. They are electronic versions of documents that stand-alone as individual files and processed accordingly as individual documents. They include the full slate of Microsoft Office documents as well as PDF files and HTML files.

Spreadsheets

Spreadsheets are of interest in electronic discovery because of their content – calculations, mathematical analyses, mailing lists, to-do lists, attendance rosters, and invoices are a few uses of spreadsheets. For discovery purposes, it is important to remember different worksheets can exist within the same spreadsheet.

Word Processing Documents

Word processing or text documents are generally created and revised in word processing application programs. Other than the content, the history of revisions and other metadata included in word processing documents can be of vital importance to both the requesting and producing parties.

Databases

Databases (from personal or business applications) are one of the most frequently sought after and disclosed sources of electronic information. Databases can contain electronic information such as financial information, electronic messages, job classifications, sales numbers, or customers. Generally, database information can be imported directly into AD Summation.

Metadata

Electronic data files contain what is commonly referred to as *metadata*, i.e., hidden or embedded data. Metadata is additional, and often valuable, information about the electronic data, but it does not appear on a printed copy of the electronic file. Computers of all types generate hidden data that is embedded in software files. Metadata is found in email messages, word processing documents, spreadsheets, and other computer files. In word processing documents, metadata may include prior revisions, revision dates, authors, and other information. Email metadata may identify who was sent a blind copy of a message, which computer created or generated a message, and who opened and viewed a message.

Other Types of Electronic Information

Other types of electronic information that you may encounter include:

•   Graphics, such as JPEG or TIFF files

•   Presentations, such as Microsoft PowerPoint files

•   Internet content, such as chat room conversations, newsgroup and listserv data, Web pages, cookie files, and Internet history logs

•   Messaging, such as text messages

The EDD Process Workflow

This section describes the electronic data discovery (EDD) workflow in which your client may be involved, from the point that a service bureau receives the data from the client to the point that the data is delivered back to the client. Understanding your client's needs and the desired end result (as well as special characteristics of EDD) is extremely important, especially when taking into consideration the various options AD Summation provides for handling EDD and the variety of ways clients choose to accept or disclose electronic data.

It is vital to understand the workflow concepts and terminology related to the processing of electronic information. A significant transition from traditional paper evidence to digital evidence continues to occur. Electronic information is now the norm for businesses and individuals, and, more significantly, much electronic information is no longer printed. Many of your clients will be increasing their use of AD Summation for searching and analyzing this electronic information to support their clients' cases. However, many do not understand the EDD process workflow, so your expertise will be invaluable to ensure a successful outcome.

The Discovering or Requesting Party

The discovering or requesting party is the party collecting (or attempting to collect) information from the opposing party. This section details the process through which a requesting party gathers electronic information from the opposition and delivers it to the service bureau for processing.

The *discovering*/requesting party must do the following:

- Determine the types of electronic information relevant to the case.
- Determine the type and location of media – servers, laptops, back-up media, and others – that will be potential sources of relevant electronic information.
- Determine the scope and estimated cost of obtaining relevant electronic information targeted.
- Determine whether data disclosed is in compliance with the request.
- Obtain electronic information from opposing party.
- Process or load the data into AD Summation.
- Search for and analyze "smoking guns" and other relevant data.
- Cull, organize, and classify the data according to issues, witnesses, and other categories.
- Convert data to an acceptable deposition or trial presentation format. Your client may need the EDD material available in a paper, TIFF, or PDF format for use in depositions and trial.

The Disclosing or Producing Party

The *disclosing* or *producing* party is the party that must respond to a request for electronic data submitted by the opposition. The producing party is confronted with common problems associated with the production of any evidence: limited time and escalating cost. This section outlines the steps generally taken by an attorney or firm in the production of electronic data.

The producing party must do the following:

- Confer with the business or individual client regarding the identification, availability, and location of requested electronic data.

- Obtain and convert data to appropriate formats (paper, TIFF, PDF, and/or native files) in order to organize and review prior to disclosure.

- Process or load the data into AD Summation.

- Cull or reduce the data set obtained from the client by eliminating irrelevant data and protecting privileged information, such as attorney-client material or trade secrets.

- Redact privileged material.

- Produce the data to the requesting party in an agreed-upon format (such as paper, TIFF, PDF, and/or native). After reviewing the data, the producing party must produce the data to the opposing party.

  – If you delivered the electronic data to your client using the native electronic documents or email, then the client can produce the electronic documents in native and/or TIFF or PDF format.

Note: To redact privileged material without contaminating the evidence, electronic documents, email messages, and attachments are typically converted to image formats. There is no entirely effective way to redact an original electronic document in native format. A native EDD file cannot be redacted and then disclosed in a native format without changing the file contents. An EDD file, such as an electronic document or an email message and attachments, must be converted from native to TIFF image format, and then redacted. The resulting redacted TIFF image then would be the version disclosed to the opposing party.

– If you delivered the electronic data to your client as text or images created from the native electronic documents, then your client cannot disclose the EDD material in its native file format. The EDD must be produced in paper or image format, with relevant metadata and full-text if requested.

Delivering EDD for Use in AD Summation

If the electronic files are presented to you in native file format, then you must extract the data and metadata from the file for loading into AD Summation.

Tips for Converting Electronic Data

Service bureaus must crack email archive files, such as the Microsoft Outlook personal folder (.PST) or the Lotus Notes folder file (.NSF), to access the email metadata that will be loaded into the AD Summation **Core Database**. For the simplest processing that uses only text and images, the EDD must be petrified, or saved to an image format, using the methodology standard to the specific service bureau. The petrification process will create the needed image files.

Keep the following things in mind when cracking email archive files:

- If the email body text is loaded into both a database field and as a full-text document in the **ocrBase**, then the client may see a search result hit from the same email message twice when searching the **Core Database** and the **ocrBase** simultaneously – one hit in the database field and a second hit in the **ocrBase** document. Both hits will be valid results, found in different renditions of the same email message. At the time of processing, an option exists to load the body as full-text only. CaseVault, Enterprise and WebBlaze's Enterprise Edition store

the body in a special OCR table text field, which eliminates the double search result issue.

- Email attachments that are email messages themselves may also have attachments of their own. The attachments to attached email messages should be processed and loaded as individual documents to ensure that the client can access and review them separately.

- The family relationships between email messages and their attachments should be preserved and reflected in the database records. For a detailed explanation of family relationships, see the section The Parent/Child (or Family) Relationship.

- The document database fields should be populated with the appropriate fielded data extracted from email messages. For example, the document database field mapped to the **TO** field on the **eMail** tab of **Defaults** dialog box (accessed from the **Options** menu) should be populated with the data that appears in the **TO** field in a Microsoft Outlook email message.

- When processing electronic documents that are not contained in an email personal folder file, metadata and full-text should also be extracted. An individual .MSG file, which is stored as a separate file on the system (not in an email personal folder file) should be treated as an electronic document (**eDoc**), not as an email message (**eMail**). The email message fielded data can be extracted and populated into the corresponding database fields used for other email messages.

Note: Many commercially available software products will, by default, handle the concerns mentioned above. If the service bureau uses an application developed in-house by them, and then they should be sure to address these points mentioned above.

Delivering in Paper or Image (TIFF or PDF) Format

Your client may decide to accept the EDD in a paper or image format for the following reasons:

• The client does not have the necessary software to search and review electronic documents in native format.

• The client finds it easier to work with the paper or image with full-text versions of the documents.

You can create a DII file that handles only the loading of the text and images into AD Summation. The basic DII file is geared toward traditional paper discovery models, in which paper documents are scanned into image formats and Optical Character Recognition (OCR) technology is used to create a full-text version. Email messages and electronic documents are received in a paper format and converted to TIFF or PDF images; the text and metadata are then extracted. In AD Summation, the image information is populated into the **ImgInfo** table, the full-text is loaded into the **ocrBase**, and the metadata is copied into the designated **Core Database** fields.

Delivering in Native Electronic File Format

If your client chooses to accept EDD from their business or individual client for production in a native file format, then you need to produce an eDII file. This style of load file is geared toward forensic-oriented service bureaus that parse metadata and email message information for loading into designated AD Summation **Core Database** fields. Native electronic files are copied to the **eDocs** repository specified in the case directory structure. This enables the use of AD Summation's multi-file format index, search, and retrieval features, and also allows users to produce electronic documents in their native formats. The eDII file also facilitates the preservation of the parent/child relationships of compound documents. For the purposes of reviewing a large EDD collection for threshold significance and winnowing, clients generally want to cull the set first and petrify later. This procedure saves the cost of petrifying large masses of irrelevant electronic information (electronic documents, email messages, and attachments). AD Summation promotes "cull first and process later" methodology. The system provides the tools to work with electronic information in native file formats and streamlines the process by which clients can bundle subsets of relevant electronic information for transport back to the service bureau. Subset processing could include running through a petrification utility for special labeling, application of Bates numbers, redaction, or other labor-intensive tasks such as dealing with the vagaries of spreadsheet files.

Tracing – Preservation of Links to Original EDD

Tracing is the process by which a database record (and its corresponding electronic document) is traced back to its native source. For example, you may have to, for disclosure, retrieve the source or native email message of an email that you converted to image format. Therefore, it is critical that you preserve the ability to trace back the image document to the source file (such as a Microsoft Outlook .PST file) that it originated from.

Service bureaus use different methods and technology to create image versions of electronic documents (email messages and email attachments) from paper. Using the method that is the standard for the service bureau, electronic documents and email messages are saved as images and all relevant text is extracted in the same way . This process will probably not be significantly different from the current methodology used by each service bureau to create image versions of EDD. When creating a DII file for loading only images and text and not native documents, the service bureau may not have to be concerned with preserving the EDD in original (or native) format. The service bureau may or may not be concerned with preserving the EDD in its original state, if the client will never be:

- Reviewing the EDD in native format
- Producing the EDD in native format
- Otherwise handling the documents in native format
- Tracing the EDD back to the source information to authenticate paper or images converted from native format

On the other hand, if any of the bullet items might be required after the initial loading of data, the service bureau must take measures to preserve the EDD in its original state. For example, suppose some months after a DII file that loaded only images and text is exported into the litigation support system, the court grants the opposition's request to have email produced as .MSG or .PST files restricted to email messages falling within the scope of the discovery request. The law firm can provide its service bureau a search result set (all database records with the email metadata) and even the petrified images associated with each search result record set. Unless the service bureau has preserved EDD in its original state (such as the .PST files containing the source from which the petrified images were obtained), the court order cannot be complied with. Moreover, not only does the EDD native source information need to be preserved, but, at the time of processing, the service bureau must be sure to relate each processed record to its native source. This process is an integral part of the working relationship between the service bureau, the litigation support software, and the client law firm. The data flow becomes seamless if the service bureau, the litigation support software vendor, and the litigation support team administering and using AD Summation work together as partners at the outset of the project.

New AD Summation DII/eDII Features

With the release of AD Summation Blaze LG Version 2.0, AD Summation established itself as a pioneer of eDiscovery management in the litigation support software industry. With the cooperation of service bureaus, AD Summation has built upon its eDiscovery technology since Version 2.0 by extending the DII file to accommodate the batch loading of electronic documents, email messages, and email attachments processed by service bureaus. Service bureaus can combine the new DII capabilities with traditional DII features to provide clients with the means to simultaneously populate their AD Summation cases with electronic documents, coded data, images, and full-text.

Some advantages introduced with the new DII functionality are:
* The ability to use relative directory paths in the DII file to indicate the location that electronic documents, images, and full-text should be loaded from. A relative path includes the latter portion of the path to a certain location or item, where the beginning portion (not explicitly stated) is relative to another location or item (such as the location of the DII file).

For example: a DII file is located on **D:\VolumeLabel** and the electronic documents are located on the CD in a subfolder named **Vol001**. The path to the electronic documents referenced in the DII file could be **\Vol001** and is relative to the location of the DII file. The DII load system will look for the electronic documents at the following path: **D:\VolumeLabel\Vol001**.

Keyword Search and Form Display

In the event that a file is not found in the specified location, a **Browse** dialog is displayed asking the user to select another relative path. This path is applied to all subsequent records until the files cannot be found again. The DII load utility then returns to browsing to the initial relative path indicated in the DII file. The user is prompted again if any files are not located.

The option to specify a full-text directory that is different than the image repository. New tokens were added to specify a full-text file directory, eliminating the requirement to copy the full-text files to the image repository before loading the DII file.

Service bureaus have the option to develop a generic DII file containing metadata from email messages and other electronic documents without specifying the exact field name that the data should be populated in. This capability provides both the service bureau and the AD Summation user the flexibility to select the database fields that data should be populated into at the time of load.

The option to include the full-text in the DII file itself, as opposed to providing separate text files for each full-text document.

Note: The relative path functionality works even if the DII file is being loaded directly from a CD. New functionality was added to allow users to load documents stored on multiple CDs, yet maintain the relative path indicated in the DII file

Note: Users can choose to delete the full-text files once they are loaded

DII/eDII File Creation

Once the DII or eDII file is created, the service bureau is ready to deliver the DII file and corresponding data to the client. The EDD, image, and full-text files should be arranged in the structure indicated in the DII file, and then copied to the media of choice.

For the purposes of this document, we will assume that the client has chosen to receive the electronic discovery documents in both native and image formats with full-text. Therefore, the service bureau will create an eDII file. For this example, a sample Microsoft personal folder (.PST) file containing two email messages will be used. One email message includes a Microsoft Excel spreadsheet attachment, and the other email message includes a Microsoft Word document and an email message attachment. The email message attached to the second email message in our example also includes a Microsoft Word document attachment, demonstrating nesting of parent/child documents. The email message and its attachments will be processed and loaded in both native and image formats.

Returning Data to the Client

On delivering the data for this eDII file to the client, the image and full-text files should be saved in one folder, the email archive files (.PST or .NSF) in a separate folder, and the extracted attachments in yet another folder. Any additional "loose" electronic documents that were not generated (extracted) from an email archive file should be saved separately and loaded using a separate DII file. Although the structure described is not required, it is a suggested best practice and simplifies using email tokens that "carry-through" from one record to the next. Loading "loose" EDD using a separate DII file prevents data contamination.

While the choices for delivery media abound, most service bureaus presently deliver processed information on CDs. Figure 1 illustrates a sample structure of contents on CDs provided by a service bureau using a AD Summation eDII file. The first CD contains two folders named **EMAIL1** and **EMAIL2**, which contain the email archive files (such as .PST or .NSF files). The second CD contains two folders named **Box001** and **Box002**, which contain the image and full-text files. The third CD contains the DII file and a folder named **Attach001**, which contains the cracked attachment files. The fourth CD contains a directory named **Attach0002**, which contains the rest of the cracked attachment files.

Note: This example is not a representation of a required or best practice folder structure, but simply an example of one acceptable structure. Another example of an acceptable directory structure is: \Volumes\Vol001\001 <DII is a sibling to the 001 directory>\<EDD files here>.
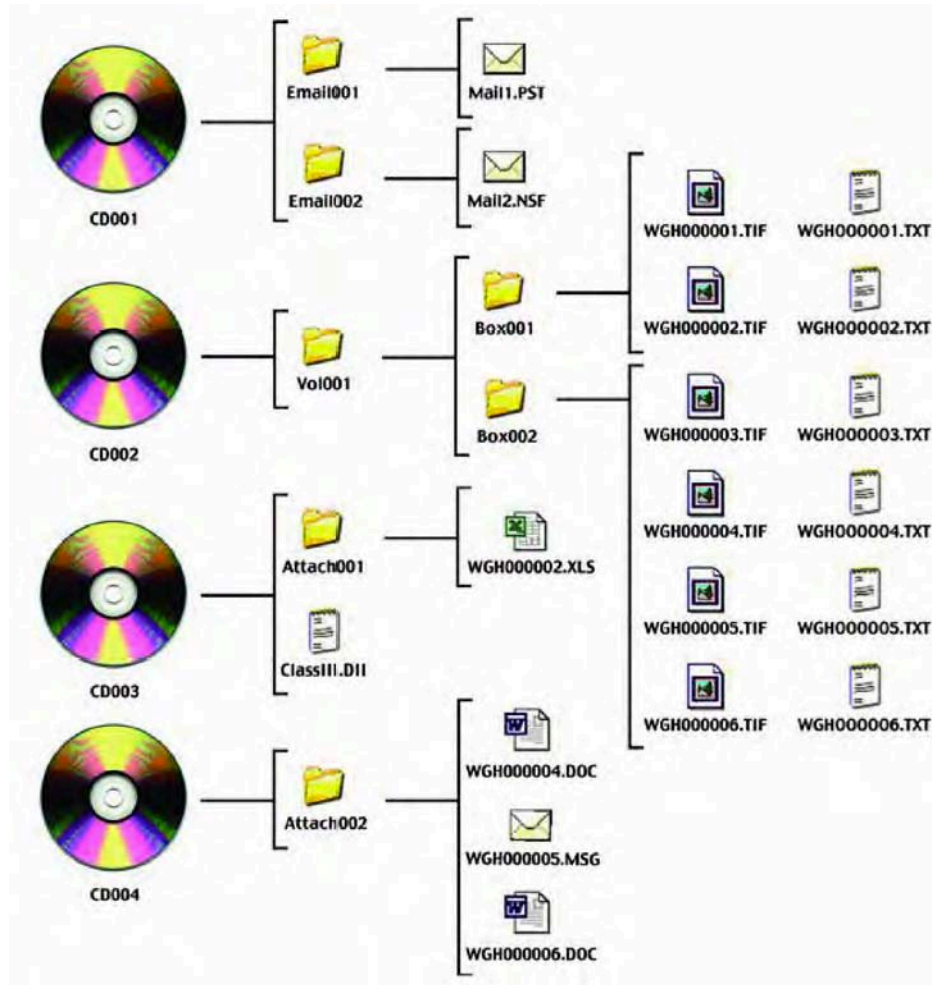
Figure 1: Sample CD Structure for loading native documents

Understanding the Structure of the eDII File

This section discusses the structure of an eDII file.

Native Electronic Document Handling

When processing and loading electronic discovery documents (EDD), AD Summation categorizes them into four distinct types: **eDocs, eMail, eMail Attachments**, and **eMail Attachments that are email messages** (and may include attachments of their own). Each EDD type requires a different set of tokens in a DII file for successful loading. These tokens initiate specific default behavior during the DII load process, which enables AD Summation to properly index and view each type of document.

eDocs (Electronic Documents)

In AD Summation, **eDocs** are evidentiary electronic documents, such as Microsoft Word documents, Microsoft Excel spreadsheets and .MSG email messages. When loaded into a AD Summation case, a database record is created for each document with the value **eDoc** added to the default **Media** field (a custom field can be used if desired). The electronic files are copied to the **eDoc** directory designated in the **Case Directory Customization** dialog box.

Note: File viewers such as QuickView Plus can display a variety of document formats (such as Microsoft Word documents, Microsoft Excel spreadsheets, or .MSG email messages) as though they are being viewed using their native applications.

AD Summation users can view an electronic document as plain text or in its native format using the application with which it was created (such as Microsoft Word), provided the application is installed on their computers. Older versions of native applications, such as Microsoft Word, may open the electronic document in a new window (rather than within AD Summation), or may not open the electronic document at all. Electronic documents that are indexed in AD Summation are fully searchable (see Electronic File Formats Supported by AD Summation's Indexer for a list of supported file types).

Electronic File Formats Supported by AD Summation's Indexer

- Adobe Acrobat (*.pdf)
- Ami Pro (*.sam)
- Ansi Text (*.txt)
- ASCII Text
- ASF media files (metadata only) (*.asf)
- CSV (Comma-separated values) (*.csv)
- DBF (*.dbf)
- EBCDIC
- EML files (emails saved by Outlook Express) (*.eml)
- Enhanced Metafile Format (*.emf)
- Eudora MBX message files (*.mbx)
- GZIP (*.gz)

- HTML (*.htm, *.html)
- JPEG (*jpg)
- MBOX email archive, including Thunderbird (*.mbx)
- MHT archives (HTML archives saved by Internet Explorer) (*.mht)
- MIME messages
- MSG files (emails saved by Outlook) (*.msg)
- Microsoft Access MDB files (*.mdb)
- Microsoft Document Imaging (*.mdi)
- Microsoft Excel (*.xls)
- Microsoft Excel 2003 XML (*.xml)
- Microsoft Excel 2007 (*.xlsx)
- Microsoft Outlook/Exchange
- Microsoft Outlook Express 5 and 6 (*.dbx) message stores
- Microsoft PowerPoint 97, PowerPoint 2000, PowerPoint 2007, and PowerPoint XP (*.pptx)
- Microsoft Rich Text Format (*.rtf)
- Microsoft Searchable TIFF (*.tiff)
- Microsoft Word for DOS (*.doc)
- Microsoft Word for Windows (all versions through Word XP) (*.doc)Microsoft Word 2003 XML (*.xml)
- Microsoft Word 2007 (*.docx)

- Microsoft Works (*.wks)
- MP3 (metadata only) (*.mp3)
- Multimate Advantage II (*.dox)
- Multimate version 4 (*.doc)
- OpenOffice 2.x and 1.x documents, spreadsheets, and presentations (*.sxc, *.sxd, *.sxi, *.sxw, *.sxg, *.stc, *.sti, *.stw, *.stm, *.odt, *.ott, *.odg, *.otg, *.odp, *.otp, *.ods, *.ots, *.odf) (includes OASIS Open Document Format for Office Applications)
- TAR (*.tar)
- TIFF (*.tiff)
- TNEF (winmail.dat files)
- Treepad HJT files (*.hjt)
- Unicode (UCS16, Mac or Windows byte order, or UTF-8)
- Windows Metafile Format (*.wmf)
- WMA media files (metadata only) (*.wma)
- WMV video files (metadata only) (*.wmv)
- WordPerfect 4.2 (*.wpd and *wpf)
- WordPerfect 5.0 and later (*.wpd and *wpf)
- WordStar versions 1, 2, 3 (*.ws)
- WordStar versions 4, 5, 6 (*.ws)
- WordStar 2000

- Write (*.wri)
- XBase (including FoxPro, dBase and other XBase-compatible formats (*.dbf)
- XML (*.xml)
- XML Paper SPecificaion (*.xps) (version 7.40)
- XSL
- XyWrite
- ZIP (*.zip)

Figure 2 shows an electronic document in its native format in AD Summation.
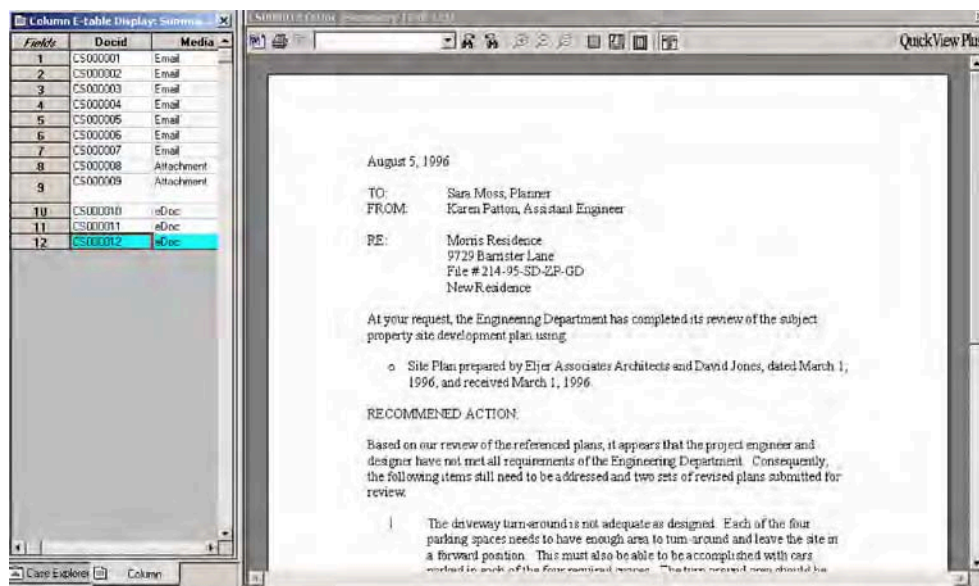
Figure 2: Viewing an Electronic Document in AD Summation in Its Native Format

Note: Files contained in ZIP files are indexed and searchable. The Search Results page displays an excerpt from the file(s) contained in the ZIP file that meet the search criteria, and reports the names of the file(s) and the ZIP file that contains them. Furthermore, the eDocs Viewer in AD Summation will not display the text of the document(s) contained in the ZIP file, but will display a message that reads **This eDoc is a ZIP file.**

Microsoft Outlook .MSG files that are located directly on a file system are treated as **eDocs** in AD Summation. **Attachments to these MSG files are not extracted as separate files or document records, but are indexed and searched along with the .MSG files.**

Email Messages

Email messages are located in email personal folder files read by applications such as Microsoft Outlook or Lotus Notes. When loaded into a AD Summation case, a database record is created for each email message, and the value **eMail** is populated into the **Media** field. The user can view email message data in the database fields in form or column display, or view the entire email message as a formatted text document.

Typical email archive file formats include Microsoft Outlook (.PST files) and Lotus Notes (.NSF files).

Figure 3 shows an email message in AD Summation.
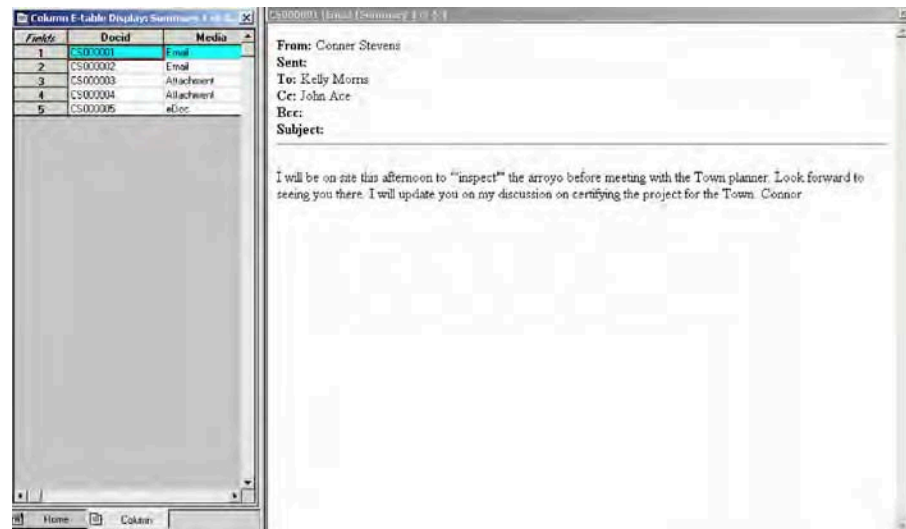
Figure 3: Viewing an Email Message in AD Summation

Email Attachments

Email attachments are electronic files such as Microsoft Word files or Corel WordPerfect files that are attached to email messages. When loaded into a AD Summation case, a database record is created for each attachment with the value **Attachment** populated into the **Media** field. Email attachment files are copied to the **email\<attachment session name>** folder in the **eDocs** case location.

Email attachments are **Blazed** when they are loaded and are searchable from the **Case Explorer**. For the purposes of search and review, an attachment is treated as an **eDoc** in AD Summation. For additional information, see the eDocs (*Electronic Documents*) section.

Email Messages as Attachments

Email attachments can be email messages themselves. When loaded into a AD Summation case, a database record is created for each attached email message with the value **Attachment** populated into the **Media** field. The attached email message is copied to the **email\<attachment session name>** folder in the **eDocs** case location.

If an attached email message includes attachments of its own, the attachments will be loaded into the AD Summation case as separate documents. The attachments will be treated as applicable, depending on whether they are **eDoc** email attachments or **eMail** email attachments.

Note: Since .MSG files located directly on a file system (not extracted from an email personal folder file) are treated as eDocs in AD Summation, all eDoc tokens apply to those files as well.

The @EDOC Token

The **@EDOC** token is used to load electronic documents (**eDocs**) that reside on a file system (such as Microsoft Word, Microsoft Excel, or Corel WordPerfect files), but are not generated or extracted from any email archive file (such as .PST or .NSF files). For a more detailed explanation of the AD Summation eDocs category, see the eDocs (*Electronic Documents*) section

When using the **@EDOC** token, the path specified can either be relative to the location of the DII file or a hard-coded path to the location of the files at the time of load. AD Summation's recommended best practice is to use a relative path. For example:

`@EDOC eDocs\ABC002_Memo.doc`

The path designated in the example above references the location of the electronic files (relative to location of the DII file) at the time of load. AD Summation then copies the files to the **efiles subfolder** in the case's **eDocs** directory and associates a database record with each file. If the document file name includes a number before it is loaded into AD Summation, you can prompt the DII load utility to remove the existing value during the load process. For information on removing numbering schemes, see The *@EDOCIDSEP Token*.

The @EATTACH Token

The **@EATTACH** token is used to load attachments to email messages that are electronic documents, such as Microsoft Word documents, Corel WordPerfect documents, and ZIP files. Attachments to email messages that are email messages themselves should not be loaded using the **@EATTACH** token, but will be treated separately with a token described in *The @ATTMSG Token* section.

During the DII load process, the attachments are copied to the **eMail** directory in the case's **eDocs** directory. AD Summation then copies the files to the case's **eDocs** directory and associates a database record with each file. If the document file name includes a number before it is loaded into AD Summation, you can prompt the DII load utility to remove the existing value during the load process. For information on removing numbering schemes, see *The @EDOCIDSEP Token* section.

The **@EATTACH** token can be used to load email messages extracted from email archive files or attachments that are part of .MSG files. Like the **@EDOC** token, the **@EATTACH** token accepts relative or hard-coded paths to the location of the files at the time of load. For example:

```
@EATTACH Attach001\WGH000004^oct 1,97 letter.doc
```

Note: Many commercially available software products will, by default, recurse and process email message attachments. If the service bureau uses an application developed in-house, then it should be sure to address the points mentioned above

The @ATTMSG Token

The **@ATTMSG** token is used to load .MSG files that are attached to email messages. The **@ATTMSG** token accepts either a relative or a hard-coded path to the location where the files are at the time of load. For example:

```
@ATTMSG Attach001\WGH000005^Subject – PGE.msg
```

When handling email message files that are attachments, a recommended best practice is to completely recurse and process them before loading so that all attachments are saved to file and ready for loading as **@EATTACH** items.

The @EDOCIDSEP Token (iBlaze only)

When loading electronic documents, the AD Summation system copies the files to the designated case's **eDocs** directory. AD Summation prepends the **DocID** number to the file name when it is loaded. Some service bureaus include a document tracking number (or other value) in the electronic document's file name before the file is loaded into AD Summation. In this case, the AD Summation DII load utility in iBlaze would prepend the **DocID** number to the existing file name, potentially duplicating the existing value (if the **DocID** and service bureau assigned tracking number are the same). For example, a service bureau assigns an electronic document the file name **FILE002_DocName.doc**. When the electronic file is loaded using the @EDOC token in the DII, the file name would, by default, be renamed to: **WGH000011^FILE002_DocName.doc**.

While AD Summation can easily handle the above document file name, users may encounter problems if they decide to produce the documents in native formats. The AD Summation **Production Tools** will remove the prefix applied at the time of load and will prepend the designated Bates number in its place. However, the AD Summation **Production Tools** will not realize that a second document tracking number appears in the file name and the document tracking number will be produced as part of the native electronic file. Such a production could make apparent gaps in the original document numbering scheme, where privilege documents were removed, and would include the internal document tracking numbers that could be considered attorney work product. If the document is produced and the Bates number assigned is **PROD000011**, then the file produced would be named: **PROD000011^FILE002_DocName.doc**.

The **@EDOCIDSEP** token is used to handle loading of electronic documents that include the document tracking number as part of the file name. The token is used to identify a delimiter used in the document file name and strip text that appears to the left of that character. In the above example, the delimiter would be the underscore (_) character. Therefore, the DII file should include the following:

`@EDOCIDSEP _`

Electronic documents loaded with a DII that references **@EDOCIDSEP_** will not include the pre-existing document number (**FILE002**), but will be renamed to the following file name: **WGH000011^DocName.Doc**. When the file is produced using the AD Summation **Production Tools**, the file name will be renamed to the following: **PROD000011^DocName.doc**.

Once referenced in the DII file, the **@EDOCIDSEP** token is applied to the record that it is referenced in and applies to all subsequent records. The argument will be applied until either the **@EDOCIDSEP** token is turned off by setting it to a blank argument (such as: **@EDOCIDSEP**), or the argument changes.

Email Message Tokens

The @MSGID Token

When loading email messages from an email archive file, the **@MSGID** token must be used and populated with the **Message ID** of the record contained in the email archive file. Email messages that are extracted from email personal folder files do not require a specific token to identify a path to the email message itself, since the email message is contained in the email archive file (the path to which should be identified in the DII file).

The @PSTFILE Token

Unlike individual email messages, an email personal folder (.PST) file must be processed by AD Summation to provide clients with the ability to view email messages in native format. The **@PSTFILE** token is used to process the .PST file by designating: 1) the location of the .PST file at the time of load, and 2) the unique ID of the .PST file. The path to the .PST file can either be hard-coded or relative to the location of the DII file at the time of load. The unique ID is recommended to be the same value assigned by the user to the .PST file when processing using AD Summation's **eDiscovery Console**, but this is not required as long as the ID names for all PST files used in the case are unique.

If either necessary value is missing, the DII load will record an error and the .PST file that corresponds to the record with the missing information will not be processed.

An example of the use of **@PSTFILE**:

```
@PSTFILE EMAIL001\PFranc.pst, PFranc_04April_2004
```

AD Summation gathers this information but does not process the .PST file until the DII load is complete. The **PST ID** (the second value) is populated into the **PST ID metadata** field as designated on the **eMail** tab in the **Defaults** dialog box (accessed from the **Options** menu) in AD Summation. The default PST ID metadata field for the default AD Summation table E-TABLE is the field STOREID. The **PST ID** argument assigned by the **@PSTFILE** token is assigned to the record it appears in and will apply to all subsequent email records. The argument is applied until either the **@PSTFILE** token is turned off by setting it to a blank argument (such as: **@PSTFILE**), or the argument changes.

@PSTFILE *must come before the* *@T* *token.* It can occur multiple times in a single DII file and assign a different argument each time. This allows the service bureau to process multiple .PST files and present the data for all .PST files in a single DII file. For example, a service bureau can process five .PST files and include five instances of **@PSTFILE** tokens with five different arguments, all in the same DII file.

The @PSTCOMMENT/@PSTCOMMENT-END Token

Users may want to record information about a .PST file that is loaded into a AD Summation case. For example, a user may want to identify where a specific .PST file came from and what it relates to (for example, client email messages related to flat space and received on April 26, 2004). The comments are associated with the .PST file designated by the **@PSTFILE** token that follows. The comments can be viewed from the email and attachment records generated from the .PST file designated in the **@PSTFILE** token.

The **@PSTCOMMENT** token should be followed by the **@PSTCOMMENT-END** token and should be listed before the **@PSTFILE** token it applies to. For example:

```
@PSTCOMMENT
<COMMENT TEXT>
@PSTCOMMENT-END
@PSTFILE EMAIL001\Pfranc.pst, Pfranc_04April_2004
```

Note: The comments will not be written to the **Core Database** record in AD Summation, but users can review the comments by right-clicking an email record and selecting the **Show PST Info** option.

Imaging

The improvements implemented in AD Summation's DII file to allow the batch loading of electronic discovery documents simply enhance pre-existing functionality. This means that image handling in a DII file remains the same as it has been in the past. The **@T** token continues to populate the **ImageTag** field in the **Imginfo** table and the **Column To Hold Image Tag** in the document database table, the **@D** token remains the reference to the location of the image repository, and the image filenames should still be listed after the **@D** token line.

An exception to the above stated rule is the single change that requires the **@T** token to be the first token of a DII record. In previous versions, the image files at the end of the record served as the indicator of the end of the current record and beginning of a new one. Since AD Summation now supports DII files that load only electronic documents without image files, the same rule can no longer apply and the image file names cannot be used to indicate the end of a record. The **@T** token is now used to indicate the beginning of a new record, which initiates the creation of a single record using the previous set of tokens (beginning with the **@T** token in the previous record).

Full-Text

Just as image functionality that existed in previous versions of AD Summation has not changed, existing full-text loading capability remains the same in the new DII structure. The **@FULLTEXT PAGE** and **@FULLTEXT DOC** tokens continue to operate as they have in the past: the first occurrence of an **@FULLTEXT** token prompts the load utility to look for and load full-text files for each subsequent DII record. AD Summation will look for either a text file for each document's page (**@FULLTEXT PAGE**) with the same file name as each image file listed with a .TXT extension, or a text file for each document (**@FULLTEXT DOC**) with the same file name as the image file for the first page of the document (but with a .TXT extension). A second appearance of the **@FULLTEXT** token with a blank argument prompts AD Summation to stop looking for and loading full-text files. Therefore, the token acts as an on/off switch for the full-text load through a DII file.

The @FULLTEXTDIR Token

The additions to the DII file include a partner to the **@FULLTEXT** tokens mentioned above: the **@FULLTEXTDIR** token. This token provides more flexibility to both the service bureau and the client when loading a DII file that includes full-text files. The **@FULLTEXTDIR** token allows the service bureau to specify a directory from which the full-text files will be copied during the load. Therefore, the full-text files do not have to be located in the same directory as the images at the time of load. The **@FULLTEXTDIR** token gives users the flexibility to load the DII file and full-text without requiring them to copy the full-text to the network first.

An example of the syntax used with the **@FULLTEXTDIR** token is:

```
@FULLTEXTDIR Vol001\Box001\ocrFiles
```

The above example shows a relative path, indicating to AD Summation that it should search for the full-text files in the same location as the DII file that is being loaded and follow any subdirectories in the **@FULLTEXTDIR** argument. The relative path works whether the DII file is on a network drive or on a CD as a sibling of the **Vol001** folder.

Just as **@FULLTEXT PAGE** and **@FULLTEXT DOC** apply to all subsequent records in the DII file until they are turned off (by adding the token after the last record that includes full-text), the **@FULLTEXTDIR** argument applies to all subsequent records in the DII file until it is changed or turned off (by including the token with a blank argument).

The @O Token

The **@O** token references the path and file name of the text file to load. The **@O** token prompts the load utility to look for and load a text file for each database record. The argument must include both the path to the full-text file at the time of load and the file name of the full-text file corresponding to the record. The **@O** token can accommodate a path relative to the location of the DII file or a hard-coded path to the location of the full-text at the time of load.

The @OCR and @OCR-END Tokens

Some service bureaus and clients prefer a different approach to loading full-text than the traditional AD Summation method of requiring the full-text to be loaded from separate ASCII text files. Some clients prefer including the full-text in the DII file itself. The **@OCR** and **@OCR-END** tokens give service bureaus the flexibility to include the full-text (including carriage returns) in the DII file. This method of loading full-text significantly improves the speed of the DII load, by eliminating the need for the system to search for and locate each text file and open it to copy the text into the **ocrBase**.

An example of the syntax used with the **@OCR** and **@OCR-END** tokens is:

```
@OCR
<full-text extracted from the electronic document, which
can span multiple lines>
@OCR-END
```

The **@OCR-END** token must appear on a separate line.

The Parent/Child (or Family) Relationship

This section describes the relationship between the documents that comprise a compound document.

The Compound Document

The default **e-Table** in AD Summation is designed to track information on a document level, as opposed to page level, which facilitates the maintenance of family relationships. Each item, whether it is an electronic document (such as a Microsoft Word file), an email message (either within an email archive file or saved as an .MSG file), or an attachment, is assigned a unique document number and separate record. This tracking methodology makes it possible for clients to code each document individually with information relevant to it. Although separate database records are created for an email message and its attachment(s), an original email message with its attachment(s) constitute a complete *compound document*. The email message that has an attachment is called the *parent document* and the attachment is called the *child document*. The parent/child (or family) relationship can be reflected in the AD Summation document database summaries. For example, when a user views an original email message in Microsoft Outlook or Lotus Notes, the attachments are an essential part of that email message. In AD Summation, although a compound document is separated into several records – one for each member that is part of the compound document – the entire compound document is considered to be one unit.

Setting Up the Parent/Child Relationship

The parent/child relationship between an email message and its corresponding attachment(s) must be reflected in the database records. AD Summation's **e-Table** tracks this relationship using the **DocID** (unique document identifier) for each document, in conjunction with the **Attchids** and **ParentID** fields. (The ATTCHIDS field in AD Summation iBlaze is called the AttachmentIDs field in AD Summation Enterprise.) The **Attchids** field must contain the **DocID**(s) that identify the attachments associated with a document. The **ParentID** field must contain the **DocID** that identifies the document's parent.

Figure 4: Email and Email Attachments Parent/Child Relationship

The DII file that the service bureau creates for the client must therefore be set up to populate the appropriate **Attchids** and **Parentid** fields.

Parent
Media: E-mail
DocID: WGH00003
Attchids: WGH000004; WGH000005

Attachment
Media: Attachment
DocID: WGH000004
ParentID: WGH00003

Attachment
Media: Attachment
DocID: WGH000005
ParentID: WGH00003
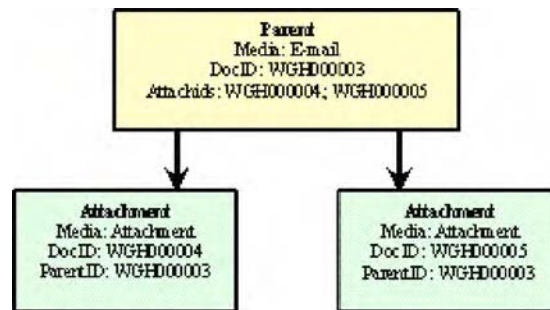
Figure 5: Diagram of Parent/Child Relationship

To facilitate the creation of the parent/child relationship, the DII file token list now includes an **@PARENTID** token and an **@ATTACH** token. When creating the email message record in the DII file, the **@ATTACH** token is used to indicate that the email message corresponding to the current record includes attachments. An example of the syntax used with the **@ATTACH** token is:

@ATTACH WGH000004; WGH000005

As you can see in Figure 5, the **@ATTACH** token can reference multiple attachments simultaneously, separated by a semicolon. Conversely, each child (attachment) record should reference the parent document using the **@PARENTID** token.

An example of the syntax used with the @PARENTID token is:

@PARENTID WGH000003

In cases where an email message has an attachment that is another email message, a record can reference both a parent and its children. Therefore, the item should reference its "immediate" parent, not the top-level parent, and the **@ATTACH** entry should reference the "immediate" children, not all of children.

For example, an email message includes an email attachment that, in turn, includes a Microsoft Word document attachment. The syntax for the parent and children records would be:

```
; Original, top-level email message:
@T WGH000003
@DOCID WGH000003
@ATTACH WGH000005

; Attached email message:
@T WGH000005
@DOCID WGH000005
@PARENTID WGH000003
@ATTACH WGH000006
; Attached Word document:
@T WGH000006
@DOCID WGH000006
@PARENTID WGH000005
```

Additional Metadata or Coded Data

The @BATESBEG and @BATESEND Tokens

As mentioned above, the AD Summation **e-Table** tracks information using a document based system by default. However, some clients may find it necessary to track all or some of the documents using a page-based system. To that end, AD Summation has added two new tokens: **@BATESBEG** and **@BATESEND**. These two tokens must be used together for the values to load successfully. In fact, if only one of the two tokens is used, the DII file load will generate errors and the **Bates Range** field will not be populated.
These tokens populate the designated values into the field designated as the **Bates Range** in the AD Summation **Default** dialog box. The following example of **Bates Range** syntax will populate **BATES000002 – BATES000004** into the designated field:

```
@BATESBEG  BATES000002
@BATESEND  BATES000004
```

Additional Metadata Tokens

Several tokens have been added to the DII load utility allowing service bureaus to easily input coded data using a DII file. The tokens do not require the service bureau to specify the database field name that the data should be populated into, and instead reference field mappings that are set in the AD Summation program. In AD Summation iBlaze, these fields mappings are selected in the **eMail** tab in the **Defaults** dialog box (accessed from the **Options** menu), and in AD Summation Enterprise, the field mappings are set in this same tab and also in the tabs "eDocs" and "Meta Columns". The data is populated into the specified fields. For example, the **@DATESENT** token populates the indicated value into the field designated as the **DATE SENT** field. For a complete list of tokens, see *Appendix A: DII Tokens*.

It is strongly recommended that service bureaus take advantage of the tokens designed specifically for coded data, as some fields require special handling that the tokens will perform by default. These tokens are designed with specific field types in mind, such as the electronic document (**@EDOC**) token that will copy the electronic document file to the **eDoc** repository and populate the relative path into the **DOCLINK** field.

The metadata tokens do not carry through to all subsequent records in the DII file, but only apply to the record in which they appear.

The @C Token

There may be situations where a client requests data to be loaded into the database fields that do not have corresponding tokens. In such a case, the service bureau can designate the field in which to populate a value by using the **@C** token.

The list of tokens is not comprehensive to the data that may be extracted from the electronic documents, however. In this case, the **@C** token should still be used and care should be taken to get the exact column name from the client for the data. The token and argument apply to the record where they first appear and carry through to all subsequent records, until it is changed or turned off (by including the token with a blank argument).

There is no limit to the number of **@C** tokens that can be included in a single DII file.

The @MULTILINE Token

In cases where a service bureau needs to populate electronic document metadata composed of multiple lines that include carriage returns, it must be able to designate the field as a multi-line field that requires special handling. This can be accomplished with the **@MULTILINE** and **@MULTILINE-END** tokens, which identify data in a DII record that includes carriage returns. The token is used to identify the field that the multi-line data will populate and the designation applies to the record that the tokens appear in and applies to all subsequent records until turned off (by including the beginning and ending tokens with a blank argument).

An example of the syntax used with the **@MULTILINE** and **@MULTILINE-END** tokens is:

```
@MULTILINE Summary
<Multi-line data inserted here>
@MULTILINE-END
```

The **@MULTILINE-END** token should always appear on a separate line. The argument will be applied until either the **@MULTILINE** token is turned off by setting it to a blank argument (such as: **@MULTILINE**), or the argument changes.

Note: If the character at the end of a line is just a line feed, then it must be changed to a carriage return line feed. There are software utilities that can automate this process. .

DII Example 1 (Email and Attachments in Native and Image Formats)

The following example uses the structure that was described in the Understanding the Structure of the DII File section of this document and applies it to the sample email archive file used in that section.

```
@PSTCOMMENTS This mail archive came from client X
Received on Friday April 9th
@PSTCOMMENTS_END
@PSTFILE PSTs\Sample.pst, SAMPLE_09April_2004

; First email record
@T WGH000001
@DOCID WGH000001
@MSGID 00000006B21E1GGFC3EGF5BCF6EFD328E23D6B815123111
@ATTACH WGH000002
@EMAIL-BODY Can you believe what this guy is asking me
to pay for???
@EMAIL-END
@TO Conner Stevens
@FROM Kelly Morris
@DATERCVD 01/06/1999
@TIMERCVD 09:04 am
@FOLDERNAME Conner Stevens — Mailbox\Deleted Items
@D @I
WGH000001.TIF
```

```
; Excel attachment to first email
@T WGH000002
@DOCID WGH000002
@PARENTID WGH000001
@EATTACH eAttach\WGH000002_Flood Damages.xls
@TO Conner Stevens
@FROM Kelly Morris
@DATERCVD 01/06/1999
@TIMERCVD 09:04 am
@APPLICATION Microsoft Excel Spreadsheet
@D @I
WGH000002.TIF
```

```
; Second email record
@T WGH000003
@DOCID WGH000003
@MSGID
000000009A12D9FFEB2DFE4ABE5DEC317D12C5A704012000
@ATTACH WGH000004; WGH000005
@EMAIL-BODY Here are some things you might want to
take a look at before you head into town for your
talk.
@EMAIL-END
@TO Conner Stevens
@FROM Kelly Morris
@DATERCVD 10/03/1997
@TIMERCVD 02:09 pm
@FOLDERNAME Conner Stevens — Mailbox\Deleted Items
@D @I
WGH000003.TIF
```

```
; MS Word attachment to second email
@T WGH000004
@DOCID WGH000004
@PARENTID WGH000003
@EATTACH eAttach\WGH000004^oct 1,97 letter.doc
@TO Conner Stevens
@FROM Kelly Morris
@DATERCVD 10/03/1997
@TIMERCVD 02:09 pm
@APPLICATION Microsoft Word Document
@D @I
WGH000004.TIF
```

```
; Data from email message attached to second email
@T WGH000005
@DOCID WGH000005
@PARENTID WGH00003
@ATTACH WGH000006
@ATTMSG eAttach\WGH000005^Subject — PGE.msg
@TO Kelly Morris
@FROM Johnny Memphis
@DATERCVD 09/28/03
@TIMERCVD 12:23 pm
@APPLICATION Microsoft Outlook
@D @I
WGH000005.TIF
```

; MS Word attachment attached to email message
attachment

@T WGH000006

@DOCID WGH000006

@PARENTID WGH000005

@EATTACH eAttach\WGH000006^Pacific gas and electric
company letter.doc

@TO Conner Stevens

@FROM Kelly Morris

@DATERCVD 10/03/1997

@TIMERCVD 02:09 pm

@APPLICATION Microsoft Word Document

@D @I

WGH000006.TIF

DII Example 2 (eDocs in Native and Image Formats)

```
@T WGH000011
@DOCID WGH000011
@MEDIA eDoc
@BATESBEG WGH000011
@BATESEND WGH000015
@EDOC An Introduction to Remote Connectivity.pdf
@FROM John Doe
@DATECREATED 10/20/2000
@DATESAVED 10/20/2000
@D @I\Vol001\
001\{\000016-000016}.tif

@T WGH000016
@DOCID WGH000016
@MEDIA eDoc
@BATESBEG WGH000016
@BATESEND WGH000016
@EDOC Flood Damages.xls
@FROM Some Guy
@DATECREATED 03/09/1999
@DATESAVED 01/24/2002
@D @I\Vol001\
001\000016.tif
```

```
@T WGH000017
@DOCID WGH000017
@MEDIA eDoc
@BATESBEG WGH000017
@BATESEND WGH000019
@EDOC Peter WoolGEOTECHNICAL CONSULTANTS.doc
@FROM eharshaw
@DATECREATED 07/13/2001
@DATESAVED 07/13/2001
@D @I\Vol001\
001\{000017-000019}.tif
```

Appendix A: DII Tokens

As of AD Summation iBlaze Version 2.5 and all versions of AD Summation Enterprise, the DII file has been extended to accommodate the loading of eDiscovery. The following table is a complete list of DII tokens, including those used for electronic documents and email messages.

A Complete List of DII Tokens- A Reference Guide
AD Summation iBlaze Product Family Version 2.9.1
AD Summation Enterprise Version 2.6

Published: 2006 | Updated: 2011

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@APPLICATION	APPLICAT	APPLICATION	The application used to view the electronic document. For example: @APPLICATION Word
@ATTACH	ATTCHIDS (Field selected for attachment Doc iDs in link fields Defaults.)	ATTACHMENTIDS (Field selected for attachment Doc iDs in link fields Defaults.)	Document IDs of attached documents. Appending the value allows the DII to populate multiple values in the field. For example: @ATTACH EML0001; EML0002
@ATTACHRANGE	ATTRANGE	ATTACHRANGE	The document number range of all attachments if more than one attachment exists. Each attachment, along with the email message, is loaded into AD Summation as its own record. The attachment range is populated with the document number of the first attachment and the last number of the last attachment. For example: @ATTACHRANGE WGH000008 – WGH0000010
@ATTMSG	MEDIA & FOLDERID	MEDIA & FOLDERID	Relative or full path and file name of the email attachment that is an email message itself. The file is copied to the msg folder (located in CaseData\ [CaseName]\eDocs\msg). The media field is populated with the value email and the FOLDERID field is coded with the session name assigned during the load of the DII file.
@AUXID	AUXID	AUXILIARYID	Useful for tracking legacy document numbers. For example: @AUXID AST00001
@BATESBEG	BATESRNG	BATESRANGE	Beginning Bates number, used with @BatesenD. For example: @BATESBEG SGD00001

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@BATESEND	BATESRNG	BATESRANGE	Ending Bates number, used with @BATESBEG. For example: @BATESEND SGD00055
@BCC	BCC	BCC	Anyone sent a blind copy on an email message. For example: @BCC Nick Thomas

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@C	Fields are specified by user.	Fields are specified by user.	Optional code used to load data into specified fields in the user's document database. This helps decrease the amount of data entry required by database users. @C is useful when the same value is repeated for a group of documents, such as documents that all have the same box number or author. The syntax for the @C token is: @C <FIELDNAME> <DATA> For example, to fill in the issues field of the database with the value mental health, the line should read: @C ISSUES Mental Health For consecutive DII records where these values are the same, you do not need to repeat the @C line. Instead, insert the next @C line in the next DII record where the data changes. To change the data value that is repeated, insert the @C line with the new value in the immediate record. To stop entering data in a field, insert an @C line with the field name following by nothing. For example: @C ISSUES The above example tells AD Summation to stop entering data in the **issues** field. **NOTE:** If a DII file record that already exists in the case contains an @C token that references a field with existing data, the data in the DII file will overwrite the data in the case.
@CC	CC	CC	Anyone copied on an email message. For example: @CC John Ace

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@D	DEFDIR (IMGINFO table)	DEFDIR (IMGINFO table)	Required token for each DII record that has an image associated with it. @D designates the directory location of the image file or files. The data specified after @D goes into the Default Directory (**Defdir**) field of the imginfo table. There are three different ways to denote the **Defdir** field: 1. @i (refers to the image location specified in Case Customize) 2. Hard coded drive letter and path or UNC path in the Defdir field (for example, F:\PFranc\Images or \\ Server\PFranc\Images). 3. @v (refers to the specified volume label of the CD- ROM or DVD) For example: @D @V CD-101:\Box_34
@DATECREATED	DATECRTD	DATECREATED	The date that the file was created, if applicable. For example: @DATECREATED 01/04/2003
@DATERCVD	DATERCVD	DATERECEIVED	Date that the file was received. For example: @DATERCVD 01/04/2003
@DATESENT	DATESENT	DATESENT	Date that the file was sent. For example: @DATESENT 01/04/2003
@DATESAVED	DATESVD	DATESAVED	When the file was saved, if applicable. For example: @DATESAVED 01/04/2003

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@DOCID	DOCID	DOCID	Document ID of a full-text document, email message, or electronic document. If the DII file includes full-text files, then the @DOCID value (instead of the @T value) is used to load and associate OCRBASE documents with the appropriate summary. For example: @DOCID EML00017
@EATTACH	DOCLINK (Field selected for linked Document field in link fields Defaults.)	DOCLINK (Field selected for linked Document field in link fields Defaults.)	Relative or full path and file name of the email attachment. The file is copied to the email directory and the relative path of the file is placed in the Doclink field. The media field is populated with the term attachment. For example: @EATTACH \\Server\Files\Flood Damages.xls
@EDOC	DOCLINK	DOCLINK	Relative or full path and file name of the electronic document. The file is copied into the efiles directory and the relative path of the file is placed in the Doclink field. The media field is populated with the term EDOC. For example: @EDOC D:\eDoc\WordDoc.doc
@EDOCIDSEP	DOCID	DOCID	This token is intended for service bureaus that use their own tracking numbers (for example, track001_Doc001.doc). The token allows AD Summation to remove the tracking ID (traCk001) from the file so that it can be replaced with a AD Summation naming convention. The token uses a one character string value to indicate the demarcation in the file name. In the example above, the underscore character separates the tracking number from the file name, so the token should be followed by the underscore character. Place this token at the top of the DII file above the individual records. For example: @EDOCIDSEP _

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@EMAIL-BODY	BODY	EMAILBODY	Body of an email message. Must be a string of text contained between @EMAIL-BODY and @EMAIL-END. The @ EMAIL-END token must be on its own line. For example: @EMAIL-BODY <E-mail message text> @EMAIL-END
@FOLDERNAME	FOLDER	FOLDER	The name of the folder that the email message comes from. For example: @FOLDERNAME Jane-Doe –Mailbox\JDoe \Inbox
@FROM	FROM	FROM	**From** field in an email message. For example: @FROM Kelly Morris
@FULLTEXT	N/A	N/A	Indicates that there are OCR documents attached to the record. The file names must match the names of the images (not including the extension), and they must be located in the same place. Variations: @FULLTEXT DOC - One full-text file exists for each database record. @FULLTEXT page - One full-text file exists for each page of the document summary. Either @FULLTEXT DOC or @FULLTEXT PAGE should be placed at the top of a DII file that contains OCR. Similar to the @C token, this statement remains in effect until turned off by using the opposite designation. For example, when using the @FULLTEXT PAGE token, turn it off by placing @FULLTEXT in the next record (above the @T line) that does not contain a full-text file.

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@FULLTEXTDIR	N/A	N/A	The @FULLTEXTDIR token is a partner to the @FULLTEXT token. This token provides more flexibility to both the service bureau and the client when loading a DII file that includes full-text files. The @FULLTEXTDIR token allows the service bureau to specify a directory from which the full- text files will be copied during the load. Therefore, the full- text files do not have to be located in the same directory as the images at the time of load. The @FULLTEXTDIR token gives users the flexibility to load the DII file and full-text files without requiring them to copy the full-text files to the network first. For example: @FULLTEXTDIR Vol001\Box001\ocrFiles The above example shows a relative path. AD Summation searches for the full-text files in the same location as the DII file that is loaded and follows any subdirectories listed after the @FULLTEXTDIR token. The relative path works whether the DII file is on a network drive or on a CD. The @FULLTEXTDIR token applies to all subsequent records in the DII file until it is changed or turned off (by including the token with a blank value).
@HEADER	HEADER	FOLDER	Email header content. The @HEADER-END token must be on its own line. For example: @HEADER <Header Text> @HEADER-END

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@I	DEFDIR (IMGINFO table)	DEFDIR (IMGINFO table)	This token is used with the @D token. The @I token refers to the image location specified in Case Customize. This location must be a drive letter and path or UNC path that points to the directory where the images are stored. AD Summation users can select any valid location or use AD Summation's default location, the IMAGES subdirectory under the Case Directory. In either case, the image files must be copied to this location.
@INTMSGID	INTMSGID	INTERNETMSGID	Internet message ID. For example: @INTMSGID <00180c34fe5$bf2d5$0500a@SKEETER>
@L	LONGNAME (IMGINFO table)	N/A	This token is optional and denotes the long name or description of the image file or files. The data after @l goes into the Longname field of the ImgInfo table. For example: @L Patient History Form **NOTE:** This token only applies to the AD Summation iBlaze product family.
@MEDIA	MEDIA	MEDIA	Populates the media field with a designated value (eDoc, eMail or Attachment). The Media field is a special field that AD Summation uses to identify whether a document is an electronic document, email, or email attachment. As a result, this token should be used with caution. For example: @MEDIA eDoc
@MSGID	MSGID	MESSAGEID	Email message ID generated by Microsoft Outlook or Lotus Notes. For example: @MSGID 000E8324B3AA800F4E954B8AA1304012000

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@MULTILINE	Any NOTE field specified.	Any LONGTEXT field specified.	Allows carriage returns and multiple lines of text to populate a specified Note field in the AD Summation iBlaze product family and a LongText field in AD Summation Enterprise. Text must be between @MULTILINE and @MULTILINE-END. The @MULTILINE-END token must be on its own line. For example: @MULTILINE FIELDNAME Here is the first line. Here is the second line. Here is the last line. @MULTILINE-END For consecutive DII records where these values are the same, do not repeat the @MULTILINE line. Instead, insert the @MULTILINE token in the next DII record where the data changes. To stop entering data in a field, insert an @MULTILINE token with the field name followed by nothing.
@NOPAGECOUNT	DOCID	DOCID	Turns off automatically using a number after a space in the document ID as the number of pages. Allows document IDs to contain spaces. Must be entered at the beginning of the DII file and applies to all records for the entire DII file. For example: @NOPAGECOUNT @T ALD00001 3602 @D @I BOX01\Dir01\ALD00001.tif If the above DII record is loaded, ALD00001 3602 appears in the document ID field. Without the @NOPAGECOUNT token, ALD00001 appears in the document ID field and 3602 populates the Pgcount field in AD Summation iBlaze and the Numpages field in AD Summation Enterprise.

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@O	N/A	N/A	There are two uses for the @O token. This token is used when the full-text documents are located someplace other than the image location as specified by the @D line of the DII file. @O tells AD Summation that there are full-text documents at a specific location. It is placed immediately below the @D line. There can only be one text file for the record, and it must have the name of the first TIFF image with a .TXT extension. The full or relative path to the full- text document must be included. For example: @O J:\docs\scanned\BK000001.txt The @O token can also be used to point to a .TXT file that does not have the same name as the TIFF file or that has no associated TIFF file.

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@OCR @OCR-END	N/A	N/A	Some service bureaus and clients prefer a different approach to loading full-text than the traditional AD Summation method of requiring the full-text to be loaded from separate ASCII text files. Some clients prefer including the full-text in the DII file itself. The @OCR and @OCR- END tokens give service bureaus the flexibility to include the full-text (including carriage returns) in the DII file. This method of loading full-text significantly improves the speed of the DII load, by eliminating the need for the system to search for and locate each text file and open it to copy the text into the OCRBASE. The @OCR-END token must appear on a separate line. For example: @OCR \<full-text extracted from the electronic document, which can span multiple lines> @OCR-END **NOTE:** When using @OCR and @OCR-END and including the full-text in the DII file, you cannot apply page breaks at specific locations in the full-text document.
@PARENTID	PARENTID (Field selected for parent iD in link fields Defaults.)	PARENTID (Field selected for parent iD in link fields Defaults.)	Parent document ID of an attachment. For example: @PARENTID WGH000003

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@PSTCOMMENT @PSTCOMMENT-END	N/A	N/A	Users may want to record information about a .PST file that is loaded into an AD Summation case. For example, a user may want to identify where a specific .PST file comes from and what it relates to. The comments are associated with the .PST file designated by the @PSTFILE token. The @PSTCOMMENT token is used in conjunction with the @PSTFILE token. @PSTCOMMENT should be followed by @PSTCOMMENT-END and must appear before the @PSTFILE token it applies to. The @PSTCOMMENT-END token must appear on its own line. For example: @PSTCOMMENT <COMMENT TEXT> @PSTCOMMENT-END @PSTFILE EMAIL001\Pfranc.pst, Pfranc_04April_2004 **NOTE:** The comments will not be written to the **Core Database** record in AD Summation, but users can review the comments by right-clicking an email record and selecting the **Show PST Information** option.

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@PSTFILE	STOREID (Field selected for PSTID field in email Defaults.)	STOREID (Field selected for PSTID field in email Defaults.)	The @PSTFILE token is used to process the .PST file by designating the following: 1) The location of the .PST file at the time of load. 2) The unique ID of the .PST file. The two values are separated by a comma. The path to the .PST file can either be hard-coded or relative to the location of the DII file at the time of load. The unique ID should be the same value assigned by the user to the .PST file when it is processed using AD Summation's eDiscovery Console. For example: @PSTFILE EMAIL001\PFranc.pst, PFranc_04April_2004 If either necessary value is missing, the DII load will record an error and the .PST file that corresponds to the record with the missing information will not be processed. AD Summation gathers this information but does not process the .PST file until the DII load is complete. The .PST file's unique ID (the second value) is populated into the PSTID field designated in email Defaults. The default field value for this field is STOREID. The PSTID value specified by the @PSTFILE token is assigned to the record it appears in and will apply to all subsequent email records. The value is applied until either the @PSTFILE token is turned off by setting the token to a blank value or the value changes. The @PSTFILE token can occur multiple times in a single DII file and assign a different value each time. This allows the service bureau to process multiple .PST files and present the data for all .PST files in a single DII file. For example, a service bureau can process five .PST files and include five instances of @PSTFILE tokens with five different values, all in the same DII file.

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@READ	READ	READ	Notes whether the email message was read. For example: @READ Y
@RELATED	OTHERIDS (Field selected for related Document IDs in link fields Defaults.)	OTHERIDS (Field selected for related Document IDs in link fields Defaults.)	The document IDs of related documents. @RELATED WGH000006
@STOREID	STOREID	STOREID	The .PST identifier. Should not be used if @PSTFILE is used. For example: @STOREID PFranc_04April_2004
@SUBJECT	SUBJECT	SUBJECT	The subject of an email message. For example: @SUBJECT Town Issues

TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@T	IMGTAG & DOCID (IMGTAG is in the IMGINFO table)	IMGTAG & DOCID (IMGTAG is in the IMGINFO table)	If @DOCID is not used, @T must be the first item listed for each database record in the DII file. This token is required for each record and designates the Image Tag. The data specified after @T goes into both the Imgtag field in the Imginfo table and the Column to hold image tag (found in imaging Defaults) in the document database table. The Image Tags must be unique. For this reason, many users choose the document number as the Image Tag. The Image Tags establish the link between the document database table and the Imginfo table. When a user clicks on a document database record and opens the image viewer to see the corresponding image, AD Summation looks at the value in the Column to hold image tag field and finds the Imginfo table record with the matching value in the Imgtag field. Then, AD Summation reads the image file location from the Imginfo table record and opens the image. For example: @T CR002931 **NOTE:** If there is a template on the Column to hold image tag field of the user's document database, then the Image Tag must conform to the template format.
@TIMERCVD	TIMERCVD	TIMERECEIVED	Time that the email message was received. For example: @TIMERCVD 11:00 a.m.
@TIMESENT	TIMESENT	TIMESENT	Time that the email message was sent. For example: @TIMESENT 10:59 a.m.
@TO	TO	TO	TO field in an email message. For example: @TO Conner Stevens

| TOKEN	BLAZE LG/IBLAZE FIELD POPULATED	ENTERPRISE FIELD POPULATED	DESCRIPTION
@TRANS	DEPOIDS (Field selected for transcript Zoom field in link fields Defaults.)	DEPOSITIONIDS (Field selected for transcript Zoom field in link fields Defaults.)	The transcript description or file name. The value populates the transcript Zoom field. For example: @TRANS Kelly Morris Vol 1
@V	N/A	N/A	This token is used with the @D token and refers to the volume label of the image location. By using a volume label instead of a drive letter, the user does not have to use the same drive letter designation for their media as had been used by the service bureau. The @V token is used most often with the images that are burned onto CD ROMs or DVDs. Substitute the volume label for the drive letter in the @D line, still including the path leading up to and including the directory in which the images are located. The user must set up Drives holding images in imaging Defaults so that AD Summation knows on which drive(s) to look for the specified volume(s). The volume label can be obtained from any drive by using the Dir command at the command prompt or by looking at the drive properties in Microsoft Windows Explorer/My Computer. When using the command prompt, the volume label will appear at the top of the directory display listing. Use the map volume to Directory option in imaging Defaults if your images are on CD-ROM or DVD, you have used @v in your DII file, and the volume label of the CD(s) is also the first subdirectory. Enabling this option tells AD Summation to map the volume label indicated after @v in the Defdir field of the Imginfo table to the drive letter(s) set in Drives holding images. For example: Defdir in ImgInfo Table: @VCD_00001: Drives Holding Images: G Maps to: G:\CD_00001\ This option is commonly used when the CDs are stored on a Meridian tower, or when the volumes have been copied to a fixed drive from a CD ROM or DVD.

Appendix B: Loading the DII File

Copying Image and Full-Text Files

In the example used in this document, the folder on the CD named **Vol001** contains image and full-text files. This volume is copied to the intended image repository location on the network. By default, this location is the **Images** subfolder that is located within the case subfolder in AD Summation. The **Case Directory Customization** dialog box identifies the default image files repository in the **Image Location**. This path is used as the default image repository when AD Summation reads the **@I** value in the **DefDir** field in the **Imginfo** table. You can view the **Case Directory Customization** dialog box by right-clicking the case name in the **Case Explorer. Error! Reference source not found** shows the **Case Directory Customization** dialog box.

Once the location to copy the images is determined, copy the **Vol001** folder there.

Note: If the DII file uses the **@FULLTEXTDIR** or **@0** tokens, then it is not necessary to copy the full-text files to the image repository. The DII load utility will look for the full-text files in the path indicated in the DII file, whether the path is relative to the location of the DII file or a full hard-coded path.
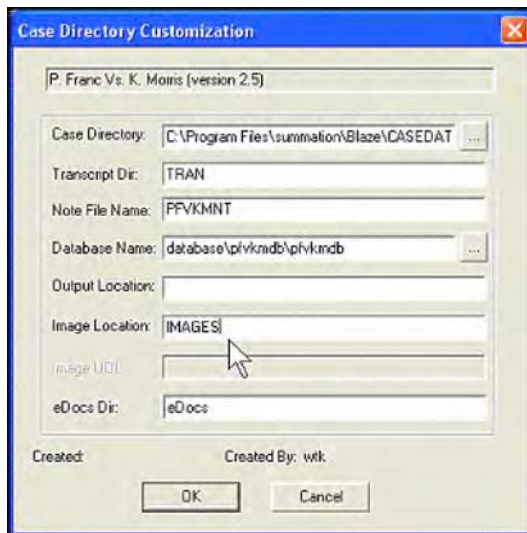
Figure 6: Determining the Default Image File location by looking at the Case Directory Customization dialog box

Figure 7 shows the default **IMAGES** repository.

Figure 7: Default Image File Location

Figure 8 shows the default **CASEDATA\IMAGES** repository with the image and full-text files copied into it. They can now be linked to corresponding database records when the user loads the DII file.
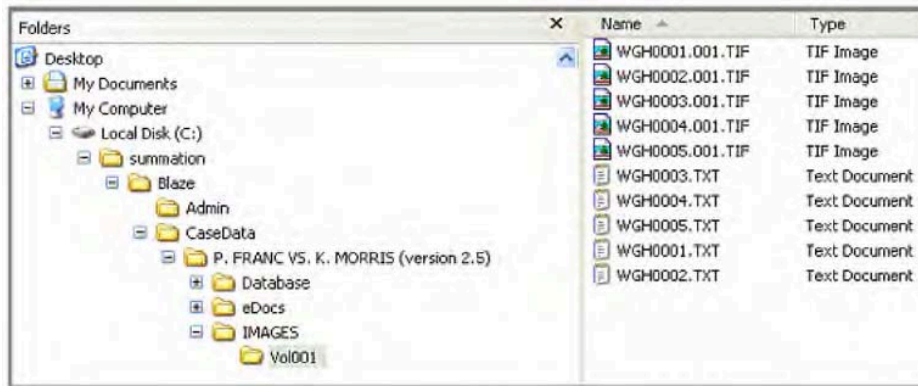
Figure 8: Copied Image and Full-Text Files

Loading the DII File

This section provides basic information for loading the DII file through the iBlaze and Enterprise applications.

To load the DII file:

1. Open the case into which you want to load the DII file.

2. With the **Case** Explorer in focus, from the **Options** menu, select **Defaults**. The **Defaults** dialog box is displayed.

3. Click the **Imaging** tab.

4. In the **Column to Hold Image Tag** menu, select the database field to hold image tags. By default, **Docid** is set.

5. Click **Load DII File**.

   The **Read DII File** dialog box is displayed.

6. Click **Browse** to locate the DII file that you want to load.

   The **Choose the DII File to Be Loaded** dialog box is displayed.

7. Select the file and click **Open**.

The **Read DII Fil**e dialog box is redisplayed, with the path and file name of the DII file shown in the box. There are two options on this dialog box.

   – **Look for eDiscovery** – Use this option if you are loading eDiscovery. This confirms that Microsoft Outlook is installed on your computer. Since native attachments will be loaded and indexed in this example, the **Look for eDiscovery** option should be selected.

    – **Delete loaded ocrBase txt files** – Use this option to delete full-text text files upon a successful load.

8. Click **OK**. The **Electronic Document Information** dialog box is displayed, with the **eDoc Session Name** box populated by default. You can keep this session name or type a new one, as well as specify a **PST ID** if needed.

9. Click **Continue.**

10. The **Saving Image Information** dialog box is displayed, showing the progress of the load. When loading is complete, the **Save to Case DB** dialog box is displayed.

11. Click **OK**.

The DII file is loaded and the database is Blazed.

Each record in the DII file may contain the **@EATTACH** token, which is followed by the relative path to the attachment file for that record. In the process of loading the DII file, each attachment is copied to the AD Summation case EDD location and the collection is indexed once loading is complete. The EDD location is designated for each case in the **Case Directory Customization** dialog box in AD Summation.

The default EDD location is the **eDocs** folder in the AD Summation case directory. Email attachments are stored in the **eMail** folder in the EDD location for each case. Each collection of attachments is organized into a folder named after the **session name** designated in the steps above. In AD Summation iBlaze, the attachment files that are copied to the EDD location are renamed using the following rule: **DOCID^file name**. For example, the file **oct 1,97 letter.doc** is copied and renamed to **EML000008^oct 1, 97 letter.doc**. This renaming of the files is unneeded in AD Summation Enterprise and thus is not performed.

Loading the .PST File (Optional)

In the final step of processing the DII file, the user is prompted to browse to and copy the .PST file that was originally cracked and processed by the service bureau. Loading the .PST file enables the user to create subset .PST files for production in native format. Whenever the DII file contains the **@PSTFILE** token, and the .PST file cannot be located (because the path relative to the DII file cannot be resolved) the user will be prompted to complete the path. The user will need to insert the CD containing the .PST file into the CD-Rom drive and browse to the appropriate folder.
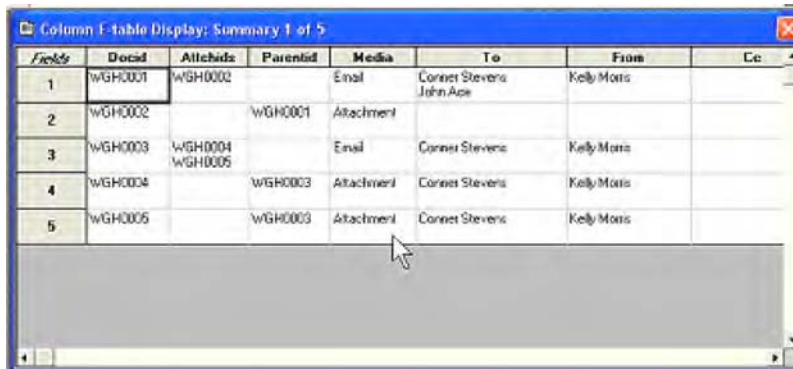
For example, if a DII file contains an **@PSTFILE** value of **PST1\mailbox.pst**, and the **PST1** folder is at the root of the CD, you should browse to the CD drive letter (such as **D:**) to complete the path.

Appendix C: Reviewing Email Messages and Attachments in a AD Summation Case

The section *Understanding the Structure of the DII File* describes how to structure the DII file, and the section *Appendix B: Loading the DII File* describes the loading process.

Once the DII file is loaded, an AD Summation **Core Database** record is created for each email message and/or attachment generated by the service bureau when it processed the email archive file. Users can search the **Core Database** and generate search results that can be expanded to include all summaries of documents related to an email message (the email message and/or its attachments) retrieved during the search. Using AD Summation's **Include Family Summaries** feature, users can display the parent email message, which contains the search hits, and its children attachments, even though they may not contain search hits. Furthermore, the parent email message and/or related attachments together make up a complete compound document and must be produced together, whether electronically or on paper.

Figure 9 shows email information displayed after it is properly loaded:

Figure 9: Viewing Email and Attachment Database Records in AD Summation

The processed image versions of email messages are linked to their corresponding records, so the documents can be reviewed in the order they would print. Moreover, segments of the accompanying extracted full-text can be annotated, issue coded, and labeled with dates. In this way, if a portion of a single document applies to one issue, and another portion applies to a different issue, the different segments can be individually set forth in a sorted report. Most importantly, the rendered components can be redacted using AD Summation's redaction tools, and Bates numbers can be sequentially applied with a separate production numbering set.

The following figure shows the docked display of the image version of the email attachment adjacent to the extracted text window. The image version (window on the left) can be redacted for production and marked up for review and analysis. The extracted full-text (window on the right) is stored in the **ocrBase**, and can be annotated, issue coded, and labeled with dates.
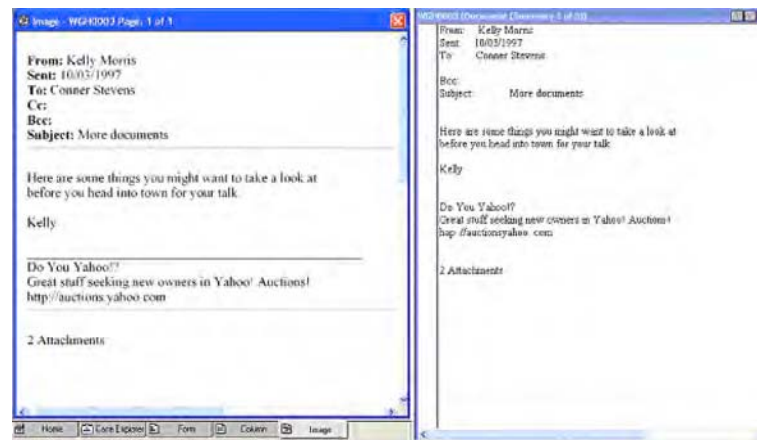

Figure 10: Image and ocrBase views of Email in AD Summation