

# NatQuery Users Guide

<b>Introduction</b> .....	<b>6</b>
<b>Software AG</b> .....	<b>6</b>
<b>NatWorks, Inc.</b> .....	<b>6</b>
<b>What the User should know</b> .....	<b>7</b>
<b>User F.A.Q. (Frequently Asked Questions)</b> .....	<b>8</b>
Building a Query .....	8
Canceling Fields when building a Query .....	8
Canceling Selection Logic.....	8
E-Mail my data.....	8
Fields to be included with returned data .....	8
Importing / Exporting Queries .....	8
Intersection .....	8
Loading ADABAS data into SQL Server or Oracle .....	8
Normalize my data – how do I make an address line in the output? .....	8
Putting my data on the web.....	9
Scheduling Query Submission .....	9
Summary Processing .....	9
Targets .....	9
Time & Date Variables.....	9
Variables – what they are and how to use them.....	10
<b>The Desktop</b> .....	<b>10</b>
<b>The Desktop</b> .....	<b>10</b>
<b>The Menu Bar</b> .....	<b>10</b>
<b>File menu</b> .....	<b>10</b>
Import.....	10
How to Import a Query .....	10
Export.....	11
How to Export a Query .....	11
<b>Edit Menu</b> .....	<b>11</b>
<b>Administration Menu</b> .....	<b>12</b>
NatQuery Configuration .....	12
User Identification Tab .....	13
Environment Paths Tab.....	13
General Defaults Tab .....	14
DWH Options Tab .....	15
License Keys Tab.....	16
Any2XML Configuration Tab .....	16
Exiting the NatQuery Configuration window.....	16
Environment Configuration .....	16
DDMs / FDTs .....	16
View Local DDMs.....	16

# NatQuery Users Guide

How to Select a DDM for Viewing .....	16
Import Environment Configuration .....	17
How to Import an Environment Configuration.....	17
<b>Contents .....</b>	<b>17</b>
<b>About NatQuery.....</b>	<b>17</b>
<b>The Toolbar .....</b>	<b>18</b>
<b>The first icon, “New Query” .....</b>	<b>18</b>
How to Select a File.....	18
<b>The second icon, “Open Query” .....</b>	<b>18</b>
Opening a Query.....	19
Deleting a Query.....	19
<b>The third icon, “Close Query” .....</b>	<b>19</b>
<b>The fourth icon, “Save” .....</b>	<b>19</b>
<b>The fifth icon, “Query Description” .....</b>	<b>19</b>
How to Use Query Description.....	20
<b>The sixth icon, “Query Variables” .....</b>	<b>20</b>
How to use the List Variables window .....	21
How to Delete a Variable.....	21
<b>The seventh icon, “Send To Server” .....</b>	<b>21</b>
Extract Type Frame (targets) .....	21
Scheduled Submission .....	23
Recurring Field Handling Options Frame.....	23
Server Options Frame .....	23
Output Handling Options Frame.....	23
Retrieval Limit Options Frame .....	24
Data Retrieval Options Frame .....	24
<b>The eighth icon, “List Program” .....</b>	<b>25</b>
<b>How to List a Program .....</b>	<b>25</b>
Extract Type.....	25
Concatenate/Flatten Option .....	26
List Options.....	26
General Options .....	27
Retrieval Limit Options .....	27
Data Retrieval Options.....	27
All Records .....	28
Intersection Records.....	28
<b>The ninth icon, “Check Server” .....</b>	<b>28</b>
General Notes on Check Server Function.....	28
<b>How to use the Check Server window.....</b>	<b>29</b>
Check Server for Update.....	30
Retrieve Request Output.....	30
Clear Selected Request .....	30
Exiting the Check Server Window .....	31
<b>Building a Simple Query .....</b>	<b>31</b>

# NatQuery Users Guide

How to Use Selection Logic .....	32
<b>Help for users .....</b>	<b>32</b>
Entering Selection Logic.....	33
Modifying/Deleting Selection Logic .....	34
Advanced Query Save.....	34
<b>Building a Dynamic Variable.....</b>	<b>34</b>
<b>Building an End-User Extraction .....</b>	<b>34</b>
<b>Building a Datawarehouse – ETL Integration .....</b>	<b>35</b>
<b>Solving an Inventory/Sales Date Range Problem .....</b>	<b>35</b>
<b>End User Windows .....</b>	<b>36</b>
<b>Select File .....</b>	<b>36</b>
How to Select a File.....	37
(Window) Query Selected Files.....	37
How to Use the Query Selected Files Window .....	37
<b>Select Fields From FileName .....</b>	<b>37</b>
To Select Fields for extraction.....	38
Deleting Fields from a Query .....	38
Search function .....	39
How to Use the Search for Field function .....	39
Occurs function.....	39
How to Specify Field Occurrences .....	40
Specifying Table Occurrences .....	40
<b>Query Selected Field.....</b>	<b>41</b>
How to Use the Query Selected Fields Window .....	41
Re-sequencing FieldName order.....	42
<b>Selection Logic.....</b>	<b>42</b>
<b>Defining a Constant Variable .....</b>	<b>42</b>
<b>Define List Variable.....</b>	<b>43</b>
How to use the Define List Variable window.....	43
Exiting the Define List Variable window .....	44
<b>Defining a Field Value .....</b>	<b>44</b>
<b>Define Variable - Variable Handling .....</b>	<b>44</b>
<b>How to Define a New Variable.....</b>	<b>45</b>
Include with Extracted Fields .....	45
<b>Define Redefinition variablename .....</b>	<b>45</b>
Redefinition Variable.....	45
<b>Define Dynamic variablename.....</b>	<b>46</b>
How to Define a Dynamic Variable.....	46
<b>Define Expression variablename .....</b>	<b>46</b>
How to Define an Expression Variable .....	47
Conditionally Assigned.....	47
Define Initialization Condition for Variablename window .....	47

# NatQuery Users Guide

To set an Initialization Condition .....	47
<b>Define Compress variablename .....</b>	<b>48</b>
How to Create a Compress Variable .....	48
<b>How to Modify a Variable Definition .....</b>	<b>49</b>
<b>Deleting Selection Logic .....</b>	<b>49</b>
<b>Windows .....</b>	<b>50</b>
<b>Window Concatenate / Flatten Fields .....</b>	<b>50</b>
<b>Window #Target Name Configuration Field Information .....</b>	<b>50</b>
<b>Window - ERROR message for Access Extract .....</b>	<b>51</b>
<b>Window - ERROR message for Any2XML Extract .....</b>	<b>51</b>
<b>Window – Manage Scheduled Tasks .....</b>	<b>51</b>
<b>(Window) Send To Server Option – Retrieval Limit .....</b>	<b>52</b>
<b>(Window) Generate Program .....</b>	<b>52</b>
<b>(Window) Define Query Variable .....</b>	<b>52</b>
How to Define a New Variable .....	52
How to Modify a Variable Definition .....	53
<b>Window - Enter Dynamic Value for #Name .....</b>	<b>54</b>
Entering a Dynamic Value .....	54
<b>Error Window .....</b>	<b>54</b>
Extract Type: Extract to XML on Server .....	54
<b>(Window) NatQuery Performance Check .....</b>	<b>54</b>
Extract Type: Read All Records .....	54
<b>(Window) Send Request to Server – Are You Sure? .....</b>	<b>55</b>
Extract Type: All Extracts .....	55
<b>(Window) FTP Process .....</b>	<b>55</b>
Extract Type: All Extracts .....	55
<b>Glossary .....</b>	<b>55</b>
<b>Access .....</b>	<b>55</b>
<b>ADD .....</b>	<b>55</b>
<b>All Records .....</b>	<b>55</b>
<b>And / Or Logic .....</b>	<b>56</b>
<b>Array .....</b>	<b>56</b>
<b>Append .....</b>	<b>56</b>
<b>Cardinality .....</b>	<b>56</b>
<b>Child Window .....</b>	<b>56</b>
<b>Compress Variables .....</b>	<b>56</b>
<b>Conditionally Assigned .....</b>	<b>57</b>
<b>Concatenate .....</b>	<b>57</b>
<b>Concatenate/Flatten Option .....</b>	<b>57</b>
<b>Constant Variables .....</b>	<b>57</b>
<b>DDM .....</b>	<b>58</b>
<b>Datawarehousing Extract Type (DWH) .....</b>	<b>58</b>
<b>Delete a Query .....</b>	<b>58</b>

# NatQuery Users Guide

Delimiter Option .....	58
Descriptor .....	58
Dynamic Variables.....	58
E mail option .....	59
Environment Configuration.....	59
Excel .....	59
Expression Variables .....	59
Extract Type .....	59
Favorite .....	59
Fields for Retrieval .....	59
File Variable .....	59
Filter on Flatten.....	59
File/Field Value .....	60
Flatten .....	60
Format.....	60
Help .....	60
Global .....	60
Initialization Condition for VariableName .....	60
Intersect.....	60
JCL / Script .....	61
List Variable .....	61
Lookup .....	61
Multi-Value Field (MU).....	61
NatCDC .....	61
Occurs .....	61
Offset Report .....	61
“Or” Logic .....	62
Periodic Group (PE) .....	62
Phonetic Descriptor .....	62
Platform .....	62
Pre Defined Variables.....	62
Predict Information .....	63
Primary Files .....	63
Production Request Processing .....	63
Program Name .....	63
Recurring / Repeating Fields .....	64
Redefinition Variables .....	64
Re-sequencing.....	64
Retrieval Limits.....	64
Search.....	64

# NatQuery Users Guide

<b>Standard File Relationship .....</b>	<b>64</b>
<b>Summary Processing .....</b>	<b>64</b>
<b>Super-Descriptor.....</b>	<b>64</b>
<b>Table.....</b>	<b>64</b>
<b>Target.....</b>	<b>65</b>
<b>Variable.....</b>	<b>65</b>

## Introduction

### Software AG

Software AG (SAG) is a German based company with over thirty years of ongoing development and enhancements in their ADABAS database product. The 4<sup>th</sup> generational program language developed for use with ADABAS is called Natural. These two products share a unique place in the IT world simply because of their longevity and their continued relevance in dealing with today's information management needs. ADABAS easily handles the huge volumes of data and transactional needs associated with any of today's large and small business environments.

ADABAS (Adaptable Data Base) can be found on mainframes as well as many open systems such as UNIX, or Windows. ADABAS is unique in the IT community for three reasons 1) It is one of the most widely used databases in the world 2) ADABAS is one of the most mature and sophisticated databases still in use today 3) The ADABAS approach to data management is non-relational allowing for much less maintenance in comparison to a more "modern" Relational Data Base Management Systems (RDBMS).

### NatWorks, Inc.

NatWorks began business in January of 1998, and is a privately held Corporation headquartered in Northfield, Vermont. Further information on NatWorks can be found on the [NatWorks' Website](#). Drawing on an extensive background of working with Software AG's (SAG) premier database known as ADABAS and the associated 4<sup>th</sup> generation language "Natural", we have developed products specifically dedicated to the extraction of data from your ADABAS database.

The original intent of our flagship product "NatQuery" was as a replacement for the SAG product Super Natural. Supernatural has been relied on for years for extractions against ADABAS. Super Natural lacks certain qualities that would make it "user friendly" such as the requirement for the user to have a programmer's through knowledge of Natural before using it. NatQuery was developed as a "user friendly" ADABAS extraction tool with an intuitive Graphical User Interface (GUI); NatQuery prompts the user frame by frame through the query building, submission, and data retrieval processes. NatQuery and NatCDC / NatCDCSP are governed by an Administrative configuration process that introduces your ADABAS sites' operating parameters and security measures into these tools. Individual user profiles are then configured and these can have varying amounts of permissions with regard to the extraction process and File access too.

# NatQuery Users Guide

NatWorks has achieved a worldwide presence through strategic partner alliances. As a result, NatWorks has customers around the world. Our clients span a range of organizations and interests, from the travel industry to energy production, banking to education, government to insurance, and it is our firm belief that every ADABAS site can benefit from having NatWorks products in their shop.

We invite your comments and suggestions, and constantly strive to find new ways to add even more value to our products.

Please note that the following information is generic across platforms that is to say this document is pertinent to ADABAS installed on the mainframe or on an “open system” such as UNIX, Linux, or Windows. Please contact your Administrator or NatWorks for specific questions regarding your site. In addition, users and Administrators should take full advantage of the “Help” documentation provided within the NatQuery product.

## What the User should know

NatQuery is a tool that allows the end-user to quickly access data found in ADABAS and bring that data back to their PC in an immediately usable format. Specifically developed for the formulation of queries, either ad hoc against a single file or for the mass migration of information from one database to another, NatQuery is easy to use and understand. This document is intended to provide the user with all the knowledge they will need to successfully build queries that will be specific to a given need, submit those queries, and subsequently retrieve the data to the target of their choice. Please feel free to contact your NatQuery Administrator should your specific questions not be answered within this document.

In the evolving development of NatQuery, the architects have tried to make the screens as intuitive as possible for the user. It is highly recommended that the user be familiar with the Microsoft Windows’s operating system including Excel and Access. These are just two of the “[targets](#)” that the user might wish to download their requested information to. It is hoped that many of the icons, buttons, and text messages will be familiar to the user and that they will already understand the actions needed to be performed when encountering these features in NatQuery. All of our products have Help options to provide the user with immediate explanations and in some cases tutorials that explain a particular process.

The intent of this tool is to make ADABAS data easily accessible for the user, and allow it to be immediately usable either from their own PC or through importation to other databases. NatQuery’s graphical user interface will shield the user from the complexities of writing their own programs to access data or porting it to other resources. NatQuery allows the user data “on demand”, in broad strokes (Read All Records) or minute specifics (Applied Selection Logic) without the need to know Natural (the 4<sup>th</sup> generational programming language developed by SAG for ADABAS) or any other programming languages.

# NatQuery Users Guide

## User F.A.Q. (Frequently Asked Questions)

### [Building a Query](#)

### [Canceling Fields when building a Query](#)

### [Canceling Selection Logic](#)

### [Changing the order of my fields as they will appear in the extract target - re-sequencing](#)

### [E-Mail my data.](#)

E-mailing your data to yourself or to another target is an administratively controlled Output Handling request made by the user.

### [Favorite Query List – Last Modified](#)

### [Fields to be included with returned data](#)

### [Importing / Exporting Queries](#)

### [Intersection](#)

### [Loading ADABAS data into SQL Server or Oracle](#)

Also, see the .PPT, ETL Integration, on our web site:

[http://www.natworks-inc.com/Products/Slide\\_Show/slide\\_show.html](http://www.natworks-inc.com/Products/Slide_Show/slide_show.html)

### [Normalize my data – how do I make an address line in the output?](#)

While the word “normalize” has several connotations in the world of IT, in this instance it refers to the ability to “break-out” data by occurrence. NatQuery accomplishes this by offering the option to [Concatenate](#) or [Flatten](#) that data which has multiple occurrences. In the “[Recurring Field Handling](#)” frame of the “[Send to Server](#)” window the user will be asked if they wish to “Always Concatenate”, “Always Flatten”, or “Always Ask”. These same options will be found in the [List Program](#) window under the [Concatenate/Flatten Option](#), with one of these choices being the default setting given by the administrator. If the default setting is “Always Ask” the user will need to make a choice prior to proceeding else a third window will appear prompting them for their decision on occurrence handling.

For example, with the returned data being sent to Microsoft Excel, how will the user want an extracted address to be returned?

Selecting Concatenate, the data will look like this:

Bob / Jones / 454 South Main Street / Suite 100 /PO Box 10/ City / Zip

In this example, there is one source record and there will be one written record.

Selecting Flatten, the data will look like this:

# NatQuery Users Guide

Bob / Jones /1/454 South Main Street / City / Zip  
Bob / Jones /2/Suite 100 / City / Zip  
Bob / Jones /3/PO Box 10 / City / Zip  
(↑ NatQuery generated Index number per record)

In this example, Address\_Line\_1 from the Concatenate choice will be the equivalent of all the lines that make up the Flatten choice. Please note in the Flatten example that an Index has been generated that reflects the number of occurrences, in this case one source record was read with three output records being written.

While this is only a simple example of the difference between Concatenate and Flatten, it is hoped the user will see the benefits of the availability of choice when extracting records concerning something as complex as payroll or perhaps medical records.

## [Putting my data on the web](#)

Ability to download ADABAS data and then automatically convert this data into XML format, complete with XSL templates for immediate presentation with web browsers, this is not the same as putting the retrieved information directly on the web.

## [Scheduling Query Submission](#)

It is quite probable that certain queries may need to run a relatively long time in order to extract the data asked for by the user. To facilitate the need to run queries at “off” times or to automatically and repeatedly run queries, NatQuery offers a scheduling option. In the event that a user will want their query to run once a day, twice a week, or even once every other month, NatQuery allows the user to have a full range of scheduling options for the submission of their queries. For more information regarding targets, please see the description provided under the seventh icon, [Send to Server](#)

## [Summary Processing](#)

The ability to perform “Summary Processing” of data before the actual loading of this data into a target such as an Excel spreadsheet will facilitate the retrieval of more definitive results from the ADABAS database for reporting purposes.

## [Targets](#)

NatWorks has provided both NatQuery and NatCDC with the ability to interface directly with many of today’s most popular desktop business tools. Some examples of these tools include; Crystal Reports and Microsoft’s Excel and Access packages. The ability to interface automatically with many ETL tools is also allowed, these include; IBM/Ascential Software’s WebSphere Data Stage suite of tools, Informatica’s Universal Data Services, and databases such as; DB2, SQL, and Oracle. For more information regarding targets, please see the description provided under the seventh icon, [Send to Server](#).

## [Time & Date Variables](#)

For reporting on time or date sensitive data such as inventory control or open order invoices, the user will need to create several variables. These variables will be used to build a [Selection Logic](#) expression in order to return data that falls within a range of dates.

# NatQuery Users Guide

See also [Dynamic Variables](#), [File Variables](#), and the tutorial concerning [Inventory tracking](#).

## Variables – what they are and how to use them

The benchmark of any database extract tool is its ability to allow the user to easily manipulate variables for the retrieval of specific information. NatQuery allows for every type of variable that a User might need for robust query definitions.

[Constant](#), [List](#), [Field](#), [Variable](#) ([Redefine](#), [Constant](#), [Expression](#), [Dynamic](#), [Compress](#))

## The Desktop

### The Desktop

Upon starting NatQuery, the user will first note the large open “desktop”. This desktop will be the area where various forms will appear which the User will “step through” when building any given query.

### The Menu Bar

At the top of the desktop is a menu bar containing three dropdown menu headers; File, Administer, and Help. A fourth menu, Edit, will be added when the user opens or begins to build a new query.

### File menu

The tabs found on the File menu initially allow a user to begin building a query ([New](#)) or open a window containing previously saved/imported queries ([Open](#)) that will then allow the user to open a specific query. The user can check the status of queries he has already sent to the server using the “[Check Server](#)” tab. After the user has built a query and it is open on the desktop, this menu will show more tabs. Along with New and Open, the user will now have access to Save, Save As, Delete, and Close. These tabs operate just as the user would expect from any Microsoft application. [Send To Server](#) and [List Program](#) are also present on this menu, for more information on these functions; please see the section on [Icons](#) in this document.

The Import and Export tabs bring up windows that allow the user to bring queries into their NatQuery environment or to send a query to another environment.

### Import

The Import Query function allows the End-user to import one or more Queries from another directory. This window displays all Query specifications that exist in the path specified by the End-user.

### How to Import a Query

The Query Path field displayed at the top of the window is the path that Queries will be imported into. This is the same path as the Query Path specified through the NatQuery Configuration function. This field is for display purposes only. The value of Query Path may only be changed through the [NatQuery Configuration](#) function

The Import From Directory list box and Import From Drive selection box are used in combination to allow the End-user to navigate to the logical or physical path that Queries will be imported from. Using these controls, navigate to the system path that contains the Queries to be imported.

# NatQuery Users Guide

The Queries Available to Import list box on the left side of the screen will display all Queries that exist in the End-user selected Import From Directory list box. Select the Query(s) that need to be imported by highlighting them. When all of the desired Queries have been selected, click the Import button to perform the indicated import request(s).

After all queries have imported, click Cancel or OK to close the window.

To cancel this function at any time, click Cancel to close the window

## Export

The Export Query function allows a user to export a Query(s) to another logical or physical drive. This window displays all Query specifications that exist in the Query path designated during the NatQuery Configuration function. Export Query can be used to export a single Query, or can be used to export multiple Queries at one time. This function is of particular value when trouble shooting a users query or queries to another workstation.

## How to Export a Query

The Query Path field displayed at the top of the window is the path from which Queries may be exported. This is the same path as the Query Path specified through the NatQuery Configuration function. This field is for display purposes only. The value of Query Path may only be changed through the NatQuery Configuration function.

The Export To Directory list box and Export To Drive selection box are used in combination to allow the user to navigate to the logical or physical path into which selected Queries will be exported. Using these controls, navigate to the system path that will receive the exported Queries.

The Queries Available for Export list box will display all Queries that exist in the current Query Path. Select the Query(s) that are required to be exported by highlighting them. When all of the Queries have been selected, click the Export button to perform the indicated export request(s).

After all queries have exported, click OK to close the window.

To cancel this function at any time, click Cancel to close the window.

## Edit Menu

The EDIT Menu drop-down generally controls functions that handle/manipulate the components of a Query Specification. The EDIT menu will only be displayed and available if there is a Query open on the NatQuery desktop. The following commands are found on this menu:

- File / Fields

This EDIT menu command allows for selecting / manipulating the files (and fields from these files) that will be defined for extraction by the Query. When File / Fields are selected, it will present the [Select Files](#) window, from which the End-user can select the desired file.

# NatQuery Users Guide

- Selection Logic  
This EDIT menu command allows for designating / manipulating the selection criteria for a Query. When Selection Logic is selected, the [Select Logic](#) window will be presented.
- Description  
This EDIT menu command allows descriptive information about a Query to be manipulated. Selecting the Description command will invoke the [Query Description](#) window.
- Variables  
This EDIT menu command allows variables to be created / manipulated for a Query. Selecting the Variables command will invoke the [List Variables](#) window, from which new variables can be added, or existing variables can be modified or deleted.

## Administration Menu

For most users the following section will be for educational purposes only. Your administrator will have already entered the information requested by NatQuery. Please consult with your Administrator prior to making any changes to this information.

If your Administrator has completed the setup process then NatQuery is all ready to begin processing your queries.

To skip over the following information please continue to the [Contents](#) portion of this manual.

## NatQuery Configuration

The NatQuery Configuration function makes use of “Tab” controls to organize the five categories of information that the window handles. By default, the User Identification Tab is displayed when the window is first invoked. Subsequent to this, the user may navigate through the other four categories of information by clicking on the specific “Tab” of interest.

The NatQuery Configuration function allows for the introduction of information that is specific to an individual’s use of NatQuery. This information falls into 6 categories, with these categories organized into tab controls on the NatQuery Configuration window. The categories are:

- [User Identification](#)
- [Environment Paths](#)
- [General Defaults](#)
- [DWH Options](#)
- [License Keys](#)
- [Any2XML Configuration](#)

# NatQuery Users Guide

It should be noted that in almost all cases NatQuery provides default values for the information requested. It should be further noted that in most cases the user would not want or need to change the information under any of the tabs found in the NatQuery Configuration window. Your database administrator typically supplies the information in this window and any changes to it should be brought to the attention of that administrator before applying them.

## User Identification Tab

The User Identification tab handles a single piece of information, the User ID of the individual that will be using NatQuery. This information, while not being particularly important for the generation of Natural data extract programs, becomes very important when NatQuery is allowed to interact with the remote Natural server through FTP (File Transfer Process).

It is mandatory that the User ID field be provided with a value, and this value is typically the User ID value of the NatQuery user as defined to the remote Natural server environment by the Administrator.

## Environment Paths Tab

The Environment Paths allow for the designation of three critical workstation paths that directly affect NatQuery's execution in terms of where NatQuery looks to find components as well as where NatQuery will look to place objects that it creates. These paths are described below:

- Data Download Path

This path instructs NatQuery as to where data that is extracted from the remote Natural server will be placed.

- Query Path

This path instructs NatQuery as to where user-entered Query specifications will be stored and retrieved, as well as any Natural data extract programs that are generated and saved locally.

- Environment Path

This path is extremely critical to the operation of NatQuery in that this path instructs NatQuery as to where it can find the administratively created configuration files, the files that describe the remote Natural environment that NatQuery will interact with.

Each of these three paths can be pointed to any available logical or physical drive path, including network drives.

When NatQuery is installed, each of the three paths listed above, default to subdirectories of the NatQuery install path. The values of these three paths can be changed manually, or can be changed by using the Browse button that is immediately to the right of the text boxes for each. Using any of these Browse buttons will invoke a window that will allow the user to easily navigate the available directories of all available drives.

In some cases, the user may wish to create a new folder (PC Directory) for use with any of the above listed paths. This can be easily accomplished by using the Create Folder button, which invokes a Create New Path window that will perform this task. Once a new folder is created, the user may change any of the three paths to

# NatQuery Users Guide

the newly created directory either by manually entering the path to the new directory into the appropriate path's textbox, or can utilize the appropriate Browse button.

## General Defaults Tab

The General Defaults category defines parameters that directly affect query generation.

The Default Extract Type field allows the user to define to NatQuery the types of extracts that the user will be most often generating. This designated default value will be presented to the user for acceptance or over-ride at the point in time that the user requests that NatQuery generate a Natural data-extract program either for sending to the remote server or for listing purposes. The currently available options for Default Extract Type are as follows:

- Download to PC File

This option instructs NatQuery to extract the requested data directly to a PC file.

- Download into Excel

This option instructs NatQuery to create a semi-colon delimited extract file, and after this file has been downloaded to the PC through NatQuery, NatQuery will invoke Excel so that Excel is automatically opened using the downloaded data file, complete with column headers for all download fields.

- DWH Software Extract

This option is designed for use with third party DWH Software tools such as IBM/Ascential Software's Data Stage Suite. This extract type will generate a Natural program that will handle the extraction of the requested data, it will submit this extraction request, and it will additionally optionally generate either a DSX (Data Stage eXchange file) or a CFD (COBOL File Description) file. This DSX or CFD can then be imported into the ETL tool, and will describe the layout of the extracted data file to that tool for subsequent processing.

- SourcePoint Extract

This option is designed to generate a Natural program that will integrate extracted data to the API of the Software AG product called SourcePoint.

- Download into Access

This option instructs NatQuery to generate an extract process that is designed to integrate extracted data directly into MS Access with the ability to automatically create a receiving table and then load this table, overlay an existing table, or append to an existing table.

- RDBMS Loading

This Extract Type allows extracted data to be directly integrated to several Relational Database Management Systems (RDBMS). Ask your Administrator if your environment allows for the use of this interface feature.

- ADACMP DECOMPRESS INFILE

This extract option supports the use of the ADACMP utility of ADABAS to accomplish mass data extraction, with the output of the ADACMP utility then being further processed by a NatQuery-generated Natural processing program.

This option is only available in the Administrative version of NatQuery, and is further only available when the

# NatQuery Users Guide

Natural / ADABAS platform is OS/390 or VSE.

- **Download to XML File**

This extract option allows data to be extracted onto the remote server platform as raw data, and when this data is then subsequently downloaded via NatQuery it will be converted to XML with the ability to optionally generate XSL.

- **DB2 Loader**

This extract option allows data to be extracted from ADABAS on the remote server with the immediate ability to then load this data into DB2 using the DB2 Load utility.

- **Any2XML Extract**

This extract option allows data to be extracted from ADABAS on the remote server with the immediate ability to publish this extracted data into XML through integration to Reliance Technology's Any2XML product.

- **Extract to XML on Server**

This option allows for data to be extracted from the remote Natural / ADABAS platform as raw data that is then immediately converted on the server into straight XML.

- **ADAULD Extract**

This extract option supports the use of the ADAULD utility of ADABAS to accomplish mass data extraction either directly from ADABAS or from an ADABAS backup, with the output of the ADAULD utility then being further processed by a NatQuery-generated Natural processing program.

This option is only available in the Administrative version of NatQuery, and is further only available when the Natural / ADABAS platform is OS/390 or VSE.

The Default Query Prefix field defines a three-character prefix that will be used by NatQuery to derive a unique default name for user-entered Query specifications when a query is first saved. The value "QRY" is provided by default.

The [Concatenate / Flatten Options](#) frame contains three radio buttons: Always Concatenate, Always Flatten, and Always Ask. The setting of this field will influence the default setting of this same option on either the [Send to Server](#) or [List Program](#) window when either is used to generate a Query Specification into a Natural Program. The final setting of this option instructs NatQuery on how to output recurring fields such as [MU](#) (Multi-Valued) or [PE](#) (Periodic Group) fields. See also [Advanced Query Save](#).

## DWH Options Tab

The DWH Options tab allows the definition of the default output paths being either DSX or CFD. The CFD (Cobol File Definition) option will supply most ETL packages with a sequential flat data file they will be able to interface with, and immediately use to populate your organizations datawarehouse.

The DSX option allows the user to interface with the proprietary IBM/Ascential Data Stage (Arden) suite of tools. This tab also defines the value of the ExportingTool parameter of any DSX files to be generated (this value defaults to "Arden Data Stage Export"). The path values provided here are to be the defaults used by NatQuery when generating these IBM/Ascential-specific components (the ExportingTool value can only be modified through the NatQuery Configuration window). When NatQuery has finished generating these

# NatQuery Users Guide

components a window will display the default directory paths for saving the component files, the user may override these default values for an alternative directory if they so choose.

In some cases, the user may wish to create a new folder (PC Directory) for use with any of the above listed paths. This can be accomplished by using the Create Folder button, which invokes a Create New Path window that will perform this task. Once a new folder is created, the user may change either the DSX or CFD paths to the newly created directory either by manually entering the path to the new directory into the appropriate path's textbox, or by using the appropriate Browse button to set the path.

## License Keys Tab

The License Key(s) tab will show the current license keys for the product(s) that are installed on the PC. Further, it provides a way to change license keys (both NatQuery & NatCDC), should this ever become necessary. Upgrading the present version of NatQuery or NatCDC will not invalidate the current license key. A valid NatQuery Administration key must be present for NatCDC to be functional.

## Any2XML Configuration Tab

This tab allows for the entry of options for NatQuery-Any2XML integration. These are User ID, Domain Name (or IP Address), and Default Project Name. The User ID is the user ID for logging into the Any2XML server. The Domain Name is the location of the Any2XML server (this can also be entered as an IP address). The Default Project Name is the project name that will appear in the Any2XML Information window, just before sending an Any2XML request to the NatQuery server. Pressing the Update Project List button will cause NatQuery to get a list of projects from the Any2XML server, thus allowing selection from the current list of projects.

## Exiting the NatQuery Configuration window

When all values are properly set, clicking the OK button will either store or update the file CONFIG.CFG in the [NatQuery FILES](#) subdirectory, and this action will close the NatQuery Configuration window.

To cancel this function at any time, click Cancel and this action will close the window. If the Cancel button is used, no updates will occur to the NatQuery Configuration information stored in CONFIG.CFG.

## Environment Configuration

### DDMs / FDTs

#### View Local DDMs

The View Local DDMs list box will display all DDMs that exist in the path found following these tabs; Administration/[NatQuery Configuration](#)/Environment Paths Tab/Environment Path. This path is originally set by your administrator and should not be changed by the user without administrative council. Please note that this function first determines the actual file names of the DDMs present and displays these filename values for selection as opposed to the PC file name into which any or all DDM's are actually stored.

#### How to Select a DDM for Viewing

Clicking on the View Local DDMs will bring up a window entitled "Select DDM for View".

To view a specific DDM, first select the file of interest by clicking on it and then click the OK button. This will open the View DDM for *filename* window, which will display all the descriptive information concerning this DDM.

# NatQuery Users Guide

To cancel this function at any time, click Cancel to close the window.

## Import Environment Configuration

The Import Environment Configuration function allows a NatQuery user to import a NatQuery Environment Configuration into the currently specified [Environment Path](#). Importing a properly configured Environment Configuration will allow NatQuery to then generate queries against the configuration. The users' original environment configuration will have been set by your administrator; before changing these setting please consult with your administrator.

## How to Import an Environment Configuration

The Config Path field displayed at the top of the window is the path that Environment Configuration will be imported into. This is the same path as the Environment Path specified in the NatQuery Configuration function, and is for display purposes only. The value of Environment Path may only be changed through the [NatQuery Configuration](#) function.

The Import From Path list box and Import From Drive selection box are used in combination to allow the user to navigate to the logical or physical path that Environment Configuration will be imported from. Using these controls, navigate to the system path that contains the Environment Configuration to be imported.

With the Import From Drive and Import From Path set, the 3 text fields in the Import Configuration Profile frame will display the status of the path. The check boxes next to the text fields are disabled in the end-user version as end users should only import a complete, verified environment configuration; Administrator versions allow these checkboxes to be used to selectively import Environment Configuration components. A properly configured Environment should minimally contain 1 DDM, at least 1 CFG file, and a Verification File.

Click Import to import the Environment Configuration.

To cancel this function at any time, click Cancel to close the window.

## Contents

The Help section for NatQuery is a great resource for the user as it provides not only a tutorial regarding how to use NatQuery "Defining a Simple Query"; it also provides the user with an immediate opportunity to access information regarding any one specific part of NatQuery.

For another sample of "[Building a Simple Query](#)", please refer to that section of this document

## About NatQuery

The "Splash" screen identifies the version and the usage (End-User, Demo or Administrator) of NatQuery. An Administrative version will have more capability than an End User version, and a Demo version will only allow the user to build a query but it will not allow extractions. The administrator may periodically update NatQuery; this update will not affect the users saved queries nor the environment settings, however the user may note that the version number will change on the Splash screen.

# NatQuery Users Guide

## The Toolbar



The tool bar that appears above the desktop has nine “icons” that will be of interest to the user. For the purposes of identification all the icons are shown here as “enabled” it should be noted that only the first two icons and the very last icon will be enabled at the start up of NatQuery and the third through the eighth icons will be inactive or grayed out. It is NatWorks’ hope that several of these icons should be familiar to the user as resembling those found in other applications.

### The first icon, “New Query”

The first icon is the New Query icon.

The First icon will be the place a user begins when they desire to build a new query. The New Query “[Select File](#)” window lets the user choose the file(s) from which to extract data. NatQuery allows the user to choose as many as five of these “[primary files](#)” from which to extract the desired data. These files have been made available to the user through administratively controlled functions, please see your DBA for more information, or consult the [NatQuery Installation and Operation Manual](#).

The Select File function allows the End-user to select the files from which data fields can then be extracted. Before the first file of the query is selected, the Select File window will display all files that are available for data extraction. After the first file is selected, the files that are available for selection will be limited to just those files that have a valid relationship to the currently selected files.

### How to Select a File

To select a file for extraction, the End-user may either single click (highlight) a specific filename and then click Select, or double click on the filename.

After a file has been selected, the End-user will be presented with the Select Fields from filename window, which will allow the required extraction fields to be designated.

### The second icon, “Open Query”

The second icon is the Open Query icon.

Clicking on this icon, the user is presented with a screen that lists all of their saved queries. Assuming this is the first time our user is working with NatQuery this window will contain few queries, if any. Note that columns are marked “Query”, “Favorite”, “Last Modified”, and that there is a “[Long Description](#)” box in this frame. By clicking either column header, [Favorite](#) or Last Modified, the user can sort their existing queries as they wish.

The Open Query window displays the list of all available queries currently defined to the NatQuery Query Path value designated by NatQuery Configuration function. In the left window, you will see the name and a short description of each query. In the right window, a long description of the currently selected query will be displayed assuming one has been provided for this query. For more information regarding the name, short description, and the long description please refer to icon five, [Query Description](#).

# NatQuery Users Guide

## Opening a Query

To open a query, use a single mouse click on its name and then click the OK button or double click on the query name.

Please be aware that you can only work on one query at a time. If the Open Query process is started with an existing query already open on the desktop, the End-user will be warned that continuing the Open Query process could replace the existing query. At this point, the End-user can cancel the Open Query process, or they can continue with the option of first saving the currently open query. When the Open Query process is successfully completed, the End-user is returned to the NatQuery desktop, which will now display the newly opened query for processing.

## Deleting a Query

To delete a query, select the query you wish with a single mouse click on it, and then click the Delete button (in the lower-left corner of this window). Use caution while deleting queries – once they are deleted from NatQuery, they are permanently gone. It is possible to copy (see [Exporting Queries](#)) the queries in the Query directory and save them as back-up file.

Please refer to the description of the fifth icon, "[Query Description](#)" for more information on how to modify or append to the information you find in this window.

## The third icon, "Close Query"

The third icon is the "Close Query" icon.

Clicking this icon produces a window asking if the user wishes to save their new query or the changes made to the open query. When saving changes to an existing query the user may want to rename this query so it will be more easily recognized in the future and thus preserve the original query.

For more information, please refer to the description of the fifth icon, "[Query Description](#)" and [Advanced Query Save](#).

## The fourth icon, "Save"

The fourth icon is the "Save" icon.

To save the query on the desktop the user need only click this icon. If the user is saving changes to an existing query, those changes will overlay the previous query. For more information please see [Advanced Query Save](#). If this query is a brand new query, clicking the Save icon will produce the "Query Description" window. For more information, please go to "[Query Description](#)" in this manual.

## The fifth icon, "Query Description"

The fifth icon is the "Query Description" icon.

In this window are several interactive windows that will be used to supply information found in the "Open Query" window, see the description of [icon two](#). The "Program Name" box allows the user to give / change the program name when it is saved. The "Short Description" box allows for a short description of the query and this short description will be used along with the "Program Name" when this saved query is referenced in the "Open Query" window. The "Long Description" box gives the user a chance to be more explicit in describing

# NatQuery Users Guide

what the query does. The “Favorite” box in this window allows the user to identify favorite or often used queries when they are saved. For more information regarding “[Favorite](#)“, please see the glossary at the end of this document

The Query Description function allows the End-user to view and edit the name, the short, and the long descriptions of a query.

## How to Use Query Description

When a query is saved for the first time, NatQuery automatically generates a Program Name for the query. This program name is comprised of a three-character prefix (as specified through the NatQuery Configuration function with the field [Query Name Prefix](#)), followed by a five-character, system generated, unique number for all queries that exist with that prefix. The Program Name field must be entered and can contain a maximum of eight characters. By default NatQuery utilizes the Program Name field as the PC filename of the query specification for easy End-user identification of any given query within the users system. By opening an Explorer window the user can navigate to their NatQuery Output file and immediately identify their query by the Program name, usually this file will have a “.TXT” extension such as in this example: QRY00001.TXT

The Short Description field allows for a short description of the query to be entered. This short description will be used in combination with the Program Name for helping to uniquely identify queries. When a query is saved for the first time, NatQuery will provide a default Short Description. The Short Description field has a maximum length of 59 characters and a minimum of one but it is a mandatory field.

The Long Description textbox allows the End-user to enter a detailed description of query processing. This use of this field is optional, but recommended for your convenience.

When the Query Description fields are properly set, clicking the OK button will close the function.

To cancel this function at any time, click Cancel to close the window. If this occurs, any previously saved information will be retained.

Using the favorite box will put an asterisk in the column marked “Favorites” in the “Open Query” window. In this way NatQuery can be asked to sort either by favorites on the top of the list or the bottom. See the description concerning icon two “[Open Query](#)”.

Note: When a saved query is opened, changing (just) the Program Name of the query and subsequently saving this query will have the same effect as performing a “Save As” function, in that a new query has now been created. If the user changes QRY00001 to be EMP00001 they will have the same query saved under two names.

## The sixth icon, “Query Variables”

The sixth icon is the “Query Variables” icon. Clicking here produces the “[List Query Variables](#)” window, where the user will find all currently defined variables and their values. Beyond viewing the available variables, this window allows the user to create a new variable through the Add button or having already highlighted one of the existing variables, they may click the Modify button as it will now be enabled and modify that variable in the subsequent windows. See [Modify Query Variable](#). In some instances, the user will also be able to delete a variable if it is not an integral part of the query.

# NatQuery Users Guide

## How to use the List Variables window

If the user desires to add a new variable definition, then clicking the Add button will present the user with the [Define Variable](#) window, initialized for a new variable definition.

If the user desires to modify an existing variable definition, then highlighting the variable of interest by clicking on it, and then clicking on the Modify button, will present the user with the [Modify Query Variable](#) window, initialized with the current specification for the selected variable.

The Select button is used when introducing a new variable into Selection logic such as in the example where a user would like the Field Name to represent two Fields; Last Name and First Name as an [Expression variable](#). The Select button is used in other instances such as when modifying a variable or making use of a pre-defined [Global](#) variable. Pre-defined variables will be made available through your administrator.

## How to Delete a Variable

If the user desires to delete an existing variable, then selecting the variable of interest by clicking on it and then clicking on the Delete button will delete the variable. Please note that a delete cannot be performed on any variable that is used in the definition of another list variable, or for any variable that appears in the [Query Selection Logic](#). Unlike deleting a file on MS Windows, if you delete a variable, it is gone for good unless you rebuild it.

To cancel this function at any time, click Cancel to close the window. If the List Variables window was presented to the user from the Selection Logic window, this will have the affect of not returning a variable for use in a Selection Logic statement.

For more in depth information please see the section titled [Miscellaneous Windows](#).

## The seventh icon, “Send To Server”

The seventh icon is the “Send To Server” icon. For more information regarding the forms on this page, see [Help for Users](#). Also, see [Advanced Query Save](#).

## Extract Type Frame (targets)

The Extract Type frame allows the End-user to select the type of extract program that is desired. The options available may vary, depending on the release (version) of NatQuery being used.

When the Server Options screen is first opened, the field Extract Type will default to the Query Target selected during the initial NatQuery Configuration function. In the Send to Server window, the user can change the default Query Target value or they can reset the Default value in the NatQuery Configuration/General Default window if another target is likely to be used more often.

Certain extract types will offer the user additional extraction options from the Send to Server window; most are specialized for that target value and will only be used under specific situations. For example using the PC File, Excel, or Access download extract types, offers the user a [Summary Processing](#) option for their required data.

If the Download to PC File extract type is selected, two additional options become available. The first is being able to specify a delimiter character for the PC (text) file. The second - which is only available if no delimiter

# NatQuery Users Guide

character was specified – is the ability to generate an [offset report](#) for the PC file. This offset report is laid out as a table and gives starting and ending offsets of fields within the file. The user will need to use an “Explorer” window to navigate to their Output file and look for the query program number that coincides with the query just run. This file will have the suffix of .OSR for Offset Report. This is useful when users need fixed-width fields in data files, and need to know the length (offset) of those fields.

If either the Download to Access or Download to Excel extract types were selected, a [delimiter option](#) becomes available. The default delimiter for either of these two extract types is a semi-colon (;), the user may change the default value. Care should be taken with this option, specifying a delimiter that occurs in the extracted data could cause problems when using that data with other software such as MS Access and or Excel.

If the DWH Software Extract type is selected, two options become available. The first is a proprietary option for the IBM/Ascential Data Stage line of products, DSX. The second option is to use CFD (COBOL File Definition) as the extract type. Select whichever option is appropriate for your data warehousing needs. NatQuery will generate the appropriate integration file with all the necessary information for the selected DWH solution to process the extracted data.

Using the, Create a Totals File, box during a DWH extract will return the desired data, as well as a separate totals file that will indicate the total number of records extracted. The Totals File will be found in the Output directory as designated by the NatQuery Administrator. This option can be used as a comparison function during the datawarehouse (Transformation & Loading) process. The “Totals” file will be an exact reflection of the total number of “Records Written to Extract File” in the Check Server window when the query has run and the user has clicked the “Check Server for Update” button. See the section entitled, [Check Server](#), for more information.

The RDBMS Extract type allows the user to integrate their data extraction into SQL or Oracle databases. To verify that this process is completed correctly the user will need access to the database of choice. Please see your Administrator if you have questions regarding this extraction option.

The Download to XML File extract will allow the user to generate their extraction request as XML. At the completion of the creation of the XML file the user will be asked if they wish to also have an XSL file created. These two files will be written into the Output file designated by the NatQuery Administrator during the configuration process and will enable the extracted data to be immediately available for use with most web browser tools.

The “Any2XML Extract” type will allow the user to generate a “Web Enabled” ADABAS extract. The user will need to have the proper ID configuration and corresponding 3<sup>rd</sup> party software tool in place to successfully run this extract type. Please see your Administrator or visit our web site, [www.natworks-inc.com](http://www.natworks-inc.com), for more information on this 3<sup>rd</sup> Party product.

The “Extract to XML on the Server” extract type will require a specific JCL/script configuration which can be provided by your Administrator. For more information on this extract type, please see you NatQuery Administrator.

## DB2 Loader

This extract option allows data to be extracted from ADABAS on the remote mainframe server with the

# NatQuery Users Guide

immediate ability to then load this data into DB2 using the DB2 Load utility.

## Scheduled Submission

The Scheduled Submission check box allows the user to manipulate the open query to be scheduled as a “Once Only” or repeating submission; Daily, Weekly, or Monthly, at their discretion. If the user has chosen a repeating submission then options become available to them such that they can set a specific interval for both time and date or even the specific month when this query will be submitted.

Aside from providing a user name when scheduling a task, the user must “set” their Windows password before submitting a scheduled request, please remember your User ID and Password are case sensitive.

To edit the scheduled task or to delete a scheduled task the user should click on the Administer/Scheduled Tasks drop down menu buttons. This action opens the [Manage Scheduled Task](#) window, here the user can see their presently scheduled tasks as well as Edit or Delete the scheduled tasks.

## Recurring Field Handling Options Frame

The [Concatenate/Flatten Options](#) frame contains three radio buttons: Always Concatenate, Always Flatten, and Always Ask. This option controls how NatQuery will process [Recurring Fields](#). If this option is set to Always Concatenate or Always Flatten, then NatQuery will begin the generation process without further interruption. If it is set to Always Ask, then NatQuery will invoke the Concatenate / Flatten Fields window that will allow the user to set the processing for each recurring field that the Query Specification may contain. Your Administrator set one of these options to be the default setting during the NatQuery User Configuration process. However, the user is given the option to override the default value for a given query or to reset the default value if the most often used target choice has changed from the original. See the section regarding NatQuery Configuration, [General Defaults](#) as well the section [Advanced Query Save](#), for more details.

## Server Options Frame

The Server Options frame will display two radio buttons, Low Priority Execution and High Priority Execution. This is an optional function, if the Administrator has set up this feature of NatQuery, then setting these radio button options will influence when the request will physically execute. For instance, a Low Priority Execution might only be allowed to execute at specific times of the day or night, whereas a High Priority Execution might allow for immediate execution. See your administrator for specific details regarding the use of Low and High extraction priority settings in your environment.

## Output Handling Options Frame

The Output Handling Options frame contains a combo-box that may contain additional options for how the request can be submitted, or how data will be handled after it has been extracted. These may include [e-mail](#), [Summary Processing](#), or other special options that have been enabled by the administrator. The default option is “[Production Request Process](#)”, which is the standard processing option, and the only option that will always be available to users. This may be the only option available, or there may be custom processing options provided by your administrator.

Typically, a user will see that the “Generate DDM for Output File” is grayed out. This capability is generally used by an Administrator to capture a specific extract dataset for more in-depth processing.

# NatQuery Users Guide

## Retrieval Limit Options Frame

The Retrieval Limit Options frame allows the End-user to choose whether NatQuery should impose any record retrieval limit on the generated program. If the No Limit option is selected, then an unrestricted Natural program will be generated. If the Limit option is selected, then a Limit field will be presented into which the End-user can enter a specific number for the records that may be returned. The extraction will return records up to the limit that the user specifies. The user may only enter numeric values, up to eight numerals, when entering a record limit. The Limit option is useful for testing a given extract program to insure that it is returning the proper records. If the limit specified is greater than the administratively defined maximum (if it has been set for this user), then query generation will not continue until the No Limit option is set, or the limit is set at or below the administratively defined maximum. Please note that using the “flatten” retrieval option will have a direct bearing on reaching your record limit, for more information please refer to the section regarding [Concatenate/Flatten](#) or the FAQ section [How do I Normalize My Data](#).

## Data Retrieval Options Frame

The Data Retrieval Options frame contains a check box to “Suppress Leading Zeros in Numeric Output” and two radio buttons: All Records and Intersection Records. By checking the “Suppress Leading Zeros in Numeric Output” box the leading zeros, if any, will be omitted when the data is loaded to a desired target such as Microsoft’s Excel or Access.

It should be noted that the radio button options are available only when a query contains more than one DDM, Data Definition Module – with only one DDM in a query, NatQuery will default to All Records. This option makes sense only if we look at file relationships ([cardinality](#)) that are allowed under NatQuery. NatQuery allows for files that are conditionally linked. This means that a record in one file may have no records in the other file(s) associated with it, or it may have one or more records in the other file(s) associated with it.

In the example case of EMPLOYEES and VEHICLES, the two files are linked together by the field PERSONNEL-ID. There will always be an employee record to match a record from VEHICLES, but the opposite is not true. There may be no record in VEHICLES that matches a given employee record because an employee may have no car, one car, or more than one car. When a query is built that uses both of these files, NatQuery has no way of knowing whether to extract all of the EMPLOYEES records, or just the ones that can be linked to a vehicle (or vehicles). Specifying the All Records option will cause NatQuery to extract all EMPLOYEES records, regardless of whether or not there is a VEHICLES record (or records) associated with them. Specifying the Intersection Records option will cause NatQuery to extract only those EMPLOYEES records that have at least one VEHICLES record associated with them. The default for this option is, All Records.

There is a case where the user will not be allowed to specify which option to use. That case is the one in which there is Selection Logic against more than one file in the query. In this case, [Intersection Records](#) will be used by default.

With the desired options entered, clicking the OK button of the Send to Server Options window will begin the automatic submission process of the query to the server.

To cancel the Send to Server function at any time, click Cancel to close the window. If this occurs, the Send to Server function will be canceled as well.

# NatQuery Users Guide

## The eighth icon, “List Program”

The eighth icon is the “List Program” icon. For more information on the forms in this window, see [Advanced Query save](#).

If this is a new query and it has not yet been saved, clicking the List Programs icon will first invoke the [Query Description](#) window.

The List Program Options window will be displayed when the End-user clicks the List Program icon. This action is to request a view of the generated program of the open query. The List Program window does not automatically reflect the parameters of the query being built. The List Program Options window is similar to the Send to Server Options window, and offers several options that will influence the Natural program generation. An Administrator typically uses this window if a problem is encountered with the running of a query. This window is governed by Advanced Query Save, but is independent from the Send To Server window such that the saved List Program options may be different than those found on the Send To Server window. The List Program generated Natural code does not necessarily have to reflect the Natural code being generated using the Send To Server options.

## How to List a Program

\*Please note the following sets of options are similar but not exact to those allowed in the Send to Server Option window.

## Extract Type

The [Extract Type](#) frame will allow the End-user to select the type of extract program that is desired. The options available will vary, depending on the release of NatQuery being used. If you are unable to find a needed Extract Type, please contact your NatQuery Administrator for more information.

Upon initial entry into the List Program Options screen, the extract type will default to the Query Target selected in the NatQuery Configuration function. The user can override the default extract type for this query. Please note that a single Query program can be either listed into a SourcePoint™ or Download Data to PC extract without any further Query specifications required.

If the Download to PC File extract type is selected, two additional options become available. The first is being able to specify a delimiter character for the PC (text) file. The second - which is only available if no delimiter character was specified – is the ability to generate an [offset report](#) for the PC file. This offset report is laid out as a table and gives starting and ending offsets of fields within the file. The user will need to use an “Explorer” window to navigate to their Output file and look for the query program number that coincides with the query just run. This file will have the suffix of .OSR for Offset Report. This is useful for users who need fixed-width fields in data files, and need to know the offset of those fields.

If either the Download to Access or Download to Excel extract types has been selected, a [delimiter option](#) becomes available. The default delimiter for either of these extract types is a semi-colon (;), but may be changed by the user to whatever character is desired or needed.

If the DWH Software Extract type is selected, two options become available. The first is a proprietary option for the IBM/Ascential Data Stage line of products. The second option is to use CFD (COBOL File Definition) as the extract type. Select whichever option is appropriate for your data warehousing needs. NatQuery will

# NatQuery Users Guide

generate the appropriate integration file with all the necessary information for the selected DWH solution to process the extracted data.

Using the Create a Totals File box during a DWH extract will return the desired data, as well as a separate totals file that will indicate the total number of records extracted. The Totals File will be found in the Output directory. The Output directory path is designated in the NatQuery Configuration / Environment Path. This option can be used as a comparison function during the datawarehouse (Transformation & Loading) process. The “Totals” file will be an exact reflection of the total number of “Records Written to Extract File” in the Check Server window when the query has run and the user has clicked the “Check Server for Update” button.

The RDBMS extract type will allow the user to “load” an SQL database or an Oracle database. The user will be prompted to make the distinction for which database in the “DB Name” window frame. For more information on the specifics of RDBMS loading please access the “Help” directory found on the tool bar of NatQuery.

The Download to XML File extract will allow the user to generate their extraction request as XML. At the completion of the creation of the XML file the user will be asked if they wish to also have an XSL file created. These two files will be written into the Output file as designated by the NatQuery Administrator and will enable the extracted data to immediately be presented through web browsers.

The “Any2XML Extract” type will allow the user to generate a “Web Enabled” ADABAS extract. The user will need to have the proper ID configuration and corresponding 3<sup>rd</sup> party software tool in place to successfully run this extract type.

The “Extract to XML on the Server” extract type will require a specific JCL/script configuration. For more information on this extract type, please see you NatQuery Administrator.

## \*DB2 Loader

This extract option allows data to be extracted from ADABAS on the remote mainframe server with the immediate ability to then load this data into DB2 using the DB2 Load utility.

\*This option is only available when the Natural / ADABAS platform is OS/390 or VSE.

## Concatenate/Flatten Option

The Concatenate/Flatten Options frame contains three radio buttons: Always Concatenate, Always Flatten, and Always Ask. The setting of this option controls how NatQuery will process recurring fields. If this option is set to Always Concatenate or Always Flatten, NatQuery will proceed with generation without further interruption. If set to Always Ask, then NatQuery will invoke the Concatenate / Flatten Fields window that will allow the user to set the processing for each recurring field that the Query Specification may contain. The default radio button that is selected will be the same as the option set in the NatQuery Configuration window, but the options are provided here to allow for temporary change. Selecting a different option will not reset the default, but will only be used to generate the current program. See NatQuery Configuration [General Defaults](#) for more information on changing the default setting for this value.

## List Options

The List Options frame allows the End-user to either display the generated Natural extract program in the List Program window or to send the program to the workstation's default printer. Depending on the chosen Extract

# NatQuery Users Guide

Type the user may first see a window displaying the queries extract field information. This window is used to specify field names for the target database or file type, as well as field format and length, for certain extract types (currently only MS Access and the DB2 Loader). There are two ways to modify the NatQuery-generated defaults. The first is to select the field entry (in the Database Field Information frame), and then click on the Edit button. The second is to double-click on the entry that you wish to modify. This will present the Output Field Information window, which is used to modify information pertaining to the selected field.

To continue with the request (after making all necessary changes), click the OK button.

To exit this window and cancel the current request, click Cancel.

## General Options

The General Options frame contains two checkboxes, Trace I/O Generation and Save Locally.

The Trace I/O Generation checkbox allows the End-user the option of opening the Trace I/O Generation function that describes the internal logic by which NatQuery determined the optimal I/O statements for the specific Query. The I/O generation path will reflect the chosen Extract Type and other parameters specified by the user on the List Program Options window.

The Save Locally checkbox allows the End-user to optionally save the generated program. If selected, the generated program will be saved into the path specified through Query Path, with a filename equal to the query's program name and an extension of "NSP". The default Query Path is controlled through NatQuery Configuration / Environment Path; it is advised that the user not change this path value unless instructed by the administrator.

## Retrieval Limit Options

The Retrieval Limit Options frame allows the End-user to choose whether NatQuery should impose any record retrieval limit on the generated program. If the No Limit option is selected, then an unrestricted Natural program will be generated. If the Limit option is selected, then a Limit field will be presented into which the End-user can enter a specific number for the records that may be returned. The extraction will return records up to the limit that the user specifies. The user may only enter numeric values, up to eight numerals, when entering a record limit. The Limit option is useful for testing a given extract program to insure that it is returning the proper records. If the limit specified is greater than the administratively defined maximum (if it has been set for this user), then query generation will not continue until the No Limit option is set, or the limit is set at or below the administratively defined maximum. Please note that using the "flatten" retrieval option will have a direct bearing on reaching your record limit, for more information please refer to the section regarding [Concatenate/Flatten](#) or the FAQ section [How do I Normalize My Data](#).

## Data Retrieval Options

The Data Retrieval Options frame contains a check box to "Suppress Leading Zeros in Numeric Output" and two radio buttons: "All Records" and "Intersection Records".

By checking the "Suppress Leading Zeros in Numeric Output" box, the leading zeros, if any, will be omitted when the data is loaded to a desired target such as Microsoft's Excel or Access. It should be noted that the radio button options are available only when a query contains more than one DDM – with only one DDM in a query, NatQuery will default to All Records. This option makes sense only if we look at file relationships that

# NatQuery Users Guide

are allowed under NatQuery. NatQuery allows for files that are conditionally linked. This means that a record in one file may have no records in the other file(s) associated with it, or it may have one or more records in the other file(s) associated with it.

## All Records

In the example case of EMPLOYEES and VEHICLES, the two files are linked together by the field PERSONNEL-ID. There will always be an employee record to match a record from VEHICLES, but the opposite is not true. There may be no record in VEHICLES that matches a given employee record. EMPLOYEES may have no car, one car, or more than one car. When a query is built that uses both of these files, NatQuery has no way of knowing whether to extract all of the EMPLOYEES records, or just the ones that can be linked to a vehicle (or vehicles). Specifying the All Records option will cause NatQuery to extract all EMPLOYEES records, regardless of whether or not there is a VEHICLES record (or records) associated with them. Specifying the Intersection Records option will cause NatQuery to extract only those EMPLOYEES records that have at least one VEHICLES record associated with them. The default is All Records.

## Intersection Records

There is a case where you will not be allowed to specify which option to use. That case is the one in which there is Selection Logic against more than one file in the query. In this case, Intersection Records will be used by default.

When the appropriate information is specified, clicking the OK button will finish the generation process. The generated program listing will be routed to either the List Program window for viewing or the workstation's default printer (and possibly saved locally as well). Optionally, the Trace I/O Generation window may also be displayed if requested.

To cancel this function at any time, click Cancel to close the window. If this occurs, the List Program function will also be canceled and the End-user will be returned to the NatQuery desktop.

## The ninth icon, “Check Server”

The ninth icon is the Check Server icon.

## General Notes on Check Server Function

This ninth icon will provide the user with information regarding the number of submissions the user is allowed, the number of queries currently submitted, the status of the currently submitted queries, and information with regard to queries that are ready to download.

It is through this icon that the user will retrieve the desired data back to their PC or cancel the query process altogether.

The NatQuery Administrator has defined a specific number of Active Requests that a specific user may have. The number of Active Requests assigned to a given user dictates how many individual data extraction queries the user can submit. One way to understand the number of Active Requests that is assigned to you is to think of them as “slots”. When you click [Send to Server](#), the Query is placed into the first “open” numbered slot. Through manual or automatic intervention, the request is put on the Server for Batch Processing. Clicking the Check Server window will allow the user to see if a Query is finished, if it is, the user may then instruct NatQuery to retrieve the data. Once the data is retrieved, NatQuery automatically “frees up” that slot for use

# NatQuery Users Guide

with another Request. For more information on retrieving your query, please see the section entitled [Check Server](#).

NatQuery assigns a status to each of the Active Requests a user may have. An Open status indicates that the Request is ready to have a Query placed into it for execution (the slot is empty). A Pending status indicates that the user has previously submitted a Query for execution (the slot is currently full), and this Query is pending execution. A Done status indicates that the NatQuery Server has completed the execution of a previously submitted Query, which indicates the user may now download the results, after which the status of the Request will return to, Open. A status of Failed means that an error was encountered with the submission or retrieval of the query for that slot. Placing the mouse over this slot should produce a window that will give the user information that indicates the nature of the problem – usually followed by the message: See your NatQuery Administrator.

The fundamental process that the Check Server window performs is relatively simple. On the local machine, NatQuery builds and maintains a “local log file” that contains information that pertains to all the current Requests that a user has submitted. By examining this local log file, NatQuery will know what Requests have been submitted for execution by the specific user. The local log file path is defined in the NatQuery Configuration menu selection under Environment Paths / Query Path. It is not recommended that a user alter this path unless instructed by the administrator.

On the Natural / ADABAS server machine, the user has a “remote log file” defined, and the various NatQuery Server processes add entries into this log as they process a user’s NatQuery Request. By examining this remote log file, NatQuery knows which submitted requests the Natural / ADABAS server machine has/has not executed. The remote log file path is defined in the NatQuery Configuration menu selection under Environment Paths / Environment Path. It is not recommended that a user alter this path unless by instructed by the administrator.

The Check Server function essentially works by retrieving the remote log file for the individual user through FTP, and then comparing the contents of this remote log to the local log, and then presenting the results to the user as a [report](#). From this report, the user can easily review the overall status of each of their active Requests, and if Requests are completed the user may then further download the results of these Requests to their local workstation.

## How to use the Check Server window

The General Request Information frame, at the top of the Check Server window displays specific administrative information concerning the user. The Next Request Number field shows the unique Request-ID number that NatQuery will assign to the next Request that the user submits – NatQuery uses these Request-Ids to allow the user’s “local log file” and their “remote log file” to be synchronized.

The Maximum Active Requests field shows how many User Request Files (slots) are assigned to the user, and the Status field shows whether or not the Administrator has set the user to be Active or not.

In the center of the Check Server window are two tabs that will be used to display information about Standard Requests or [Scheduled Requests](#). The information displayed in either box is presented as a report for each Request File defined to the user, please note that a user has unlimited submission slots for Scheduled Requests. By examining these reports, the user will readily see the status of their Requests.

# NatQuery Users Guide

## Check Server for Update

To determine the status of any “Pending” Requests that the user may have, click the Check Server for Update button in the lower left corner of the window. Depending on whether or not the user has utilized FTP previously in this session of NatQuery, NatQuery may respond by displaying a request for the FTP Password. If this occurs, the user must type the unique password that corresponds to their User ID; this information is critical to allowing the FTP process to work.

Clicking the Check Server for Update button causes NatQuery to retrieve the user’s “remote log file” and then internally perform the logic necessary to compare / match the user’s “remote log file” to their “local log file”. If any previously submitted Requests have now been processed, NatQuery will update and expand the displayed report to show this, and the status of any processed Request will change from “Pending” to “Done” or “Failed”. The active query slot will show; the slot number, the program name, the status, date and time submitted, date and time completed, target name, and number of records written.

Moving the mouse (rollover) over the active query slot will reveal the details of the report. Below is a sample of a rollover query submission report:

Request Slot 1 -	QRY00001
Request-ID:	539
Request Type	Data Extraction
Output Filename	joescomputer-Out1.TXT
Records Read	1107
Records Accepted	1107
Records Written	1107

Note that the Request ID is not the User’s ID at all. It is the sequential number of the submissions made by the user; the next query’s Request ID will be 540 and so on.

## Retrieve Request Output

To retrieve the results of any Request marked “Done”, the user begins by clicking on any part of the report that is related to the Request of interest. This action will highlight the entire section of the report that is related to that specific Request, and will additionally cause the Retrieve Request Output button to become enabled. Clicking on the Retrieve Request Output button will then cause NatQuery to retrieve the file, and, once the data has been downloaded, will additionally cause that request slot to show as being “Open”.

In the case of Scheduled Requests the user will be asked if they wish to delete the scheduled submission only if it is a “Once Only” submission. With an on-going scheduled submission, after retrieving the currently available data the slot will show as being Open until the scheduled submission date and time when the slot will show it as Pending.

## Clear Selected Request

In some situations, a user may wish to remove a previously submitted Query that is currently marked “Pending” from processing, I.E., “clear” a Request slot. In this case, the user begins by clicking on any part of the report that is related to the Request of interest. This action will highlight the entire section of the report that is related to that specific Request, and will additionally cause the “Clear Selected Request” button to become enabled. Clicking on the Clear Selected Request button will then cause NatQuery to clear the Request “slot”, and change

# NatQuery Users Guide

the status of this slot from “Pending” to “Open”.

In the case of Scheduled Requests the Clear Selected Request button will not be enabled unless the query has a status of Pending or Failed. The user will not be able to delete a query that is scheduled for on-going submissions in the Check Server window; they may delete the current Pending submission of the on-going submissions. Queries that are scheduled as “Once Only” may be completely deleted in the Check Server Window.

To delete any scheduled submission the user may go to the NatQuery Menu bar: Administer / Scheduled Tasks option and highlight the query in question and delete it entirely or edit it. A scheduled request may be edited or deleted at any time through this menu option.

## Exiting the Check Server Window

The user can close the Check Server window by clicking the OK button. If there are still Requests Pending or Requests that are done, the user may revisit the Check Server window at their leisure. The user may even terminate their NatQuery session, begin it again at a later time, and go directly to the Check Server icon to retrieve their results. Please note that NatQuery will not allow the user to exceed their request allotment for Standard Requests. If the user has submitted the total number of requests that they are allowed, the user must retrieve (empty) or delete at least one Standard Request before they submit another query.

To cancel the Check Server window at any time, the user may click the Cancel button. This action will cause the Check Server window to close.

## Building a Simple Query

The users’ goal is to retrieve all the employees named SMITH from the company’s ADABAS database.

Using the “[New Query](#)” icon the user must select a file presented in the Select File window by either double clicking or by highlighting the desired File and clicking the newly enabled Select button. In this example, the user chooses the EMPLOYEES File.

The user will now be presented with a window asking them to “[Select Field from FileName](#)”. Under the Possible Fields To Select column the user will double click on or highlight and use the Add button to select the field or fields they wish to use. The Add All button will add all the fields available from this File for use in the query. The available list of fields may be quite large; if this is the case, the user may wish to use the Search function. See [User windows](#) for more information regarding the [Search](#) function. In this case, the user chooses the NAME Field, and because a report repeating the name SMITH is not very useful, the user also chooses the fields: FIRST NAME, and PERSONNEL ID.

A Field(s) that has been picked for use will appear in the “[Selected Field to Retrieve](#)” column. If the user wishes to omit a choice or all the choices, they may do so using the Delete or Delete All buttons. The Occurs button will only be enabled for a highlighted Field that contains [multiple occurrences](#). The user will recognize a Field that has multiple occurrences by the parentheses after the Field name. An example might look like: MAINT- COST (ALL). For more information please review [Occurs](#), [PE’s](#) and [MU’s](#), in the Glossary at the end of this manual.

# NatQuery Users Guide

Having moved all the needed Fields into the [Selected Field to Retrieve](#) column, the user clicks the OK button and is presented with an opportunity to repeat the process in a second File. Only those files with an administratively pre-defined relationship ([cardinality](#)) with the original file will be made available during this stage with the user being allowed to select up to five files total. For the purpose of this example, let us assume the user wishes to use only one File, they would then click the Continue button found on the Select File window. This brings up the Selection Logic window; here the user will build the actual criteria that will determine the data they wish to extract.

## [How to Use Selection Logic](#)

The “[Selection Logic](#)” window has several interactive frames where the user can indicate the criteria they wish to use in their query. The upper most frame on the right side of the screen, Select Records From, contains the Files chosen to extract information from. In our example, only one File, the EMPLOYEES file, was selected and by default, this File will be the only file available to build selection logic against. When using multiple Files, the user may only build selection logic against one file at a time. In the Selection Mode frame the user has the choice to Apply Selection Logic or to Read All Records. Clicking the Apply Selection Logic button, the user will notice that the Selection Logic To Apply frame will become active, prompting the user to supply a Where Field selection. If the Read All Records button were clicked, then the Current Selection Logic frame displays the message Read-All-Records-From *FileName* and the user would be free to now submit this query. In our example, selection logic is needed and the user will click on the Apply Selection Logic button. The goal was to retrieve those employees named SMITH, the field NAME is then chosen from the Where Field column as this field represents all the employees Last Name. This action enables the, is, column. There are multiple choices in this column; in the example the user will use Equals, this query now reads “File01.NAME equals”. The, to, column, become enabled with four choices; “[Constant](#)”, “[List](#)”, “[Field](#)” and “[Variable](#)”. In the example, the user needs only the employees named SMITH, they will click the Constant variable and enter SMITH in the resulting window and click OK. This query now reads; from File01 (EMPLOYEES), return all the records where the last name is equal to SMITH. The returned data will include all the fields listed in the Select Fields to Retrieve window; NAME, FIRST NAME, and PERSONNEL ID.

When all Selection Logic has been entered, the OK button can be used to store the new Selection Logic and close the window. When the “Selection Logic” window closes, the user should note that their newly built query appears in the Query Selection Logic window, and that all the tool bar icons are now enabled on the NatQuery Desktop.

At this point in “Building a Simple Query”, the user has created a valid query and can click [icon seven](#) “Send To Server” to begin the process of retrieving the desired data.

## Help for users

For more query examples, please refer to the Help button found at the top of the open NatQuery desktop. NatQuery Help will be found by clicking the Contents tab found in the drop down menu presented after clicking the Help button. Sample queries and definitions will be found under the End Users Guide/Tutorial/Defining Queries and Generating Extracts. For term or word definitions, please use the Glossary in this manual or the Glossary located in the Help section of NatQuery. For graphical examples of how to build queries, Power Point™ shows can be found on our web site ([www.natworks-inc.com](http://www.natworks-inc.com)) under the Slide Show page; [http://www.natworks-inc.com/Products/Slide\\_Show/slide\\_show.html](http://www.natworks-inc.com/Products/Slide_Show/slide_show.html)

# NatQuery Users Guide

The Selection Logic window provides all the functionality to handle the End-user required Selection Logic for a Query. The user may modify the existing Selection Logic, or add additional logic here as well.

The user should be aware that your Administrator might have imposed limits on the number of records you will be allowed to retrieve. If you encounter irregularities in the data retrieval process such as an error message, or need advice regarding query submissions, please save the query in question and contact your Administrator.

## Entering Selection Logic

If more than one file has been selected for data extraction, then the End-user will begin by selecting one of these files to start the process of building Selection Logic for their query. For the selected file, the End-user must then decide the Selection Mode to be used. If Read All Records is specified, then this is the only Selection Logic that can be applied against the specified file. If the Apply Selection Logic mode is selected, then one or more selection statements can be entered for the specified file, as described below. After the user has built the necessary logic against the first file they can then return to the Select Records From frame and select another file to work with, all of the Files may be used to build a query but minimally there must be at least one used. If only one file has been selected for data extraction, the Select Records From selection box will be pre-selected with this file.

When using the Apply Selection Logic mode, the End-user interacts with the graphical elements in the Select Logic to Apply frame. Using these graphical elements, the End-user constructs a Selection Logic statement, which has the basic syntax of:

Field-Name with an Operator and a Value

The user sees this syntax equation as the following headers in the Select Logic to Apply frame;

“Where Field”            “is ...”            “to ...”

As the selection statement is being built, it is shown in the Current Selection Logic frame. NatQuery visually reinforces the process of building Selection Logic statements using flashing prompts. To begin a selection logic statement, the End-user first selects a *FieldName* from the specified File from the “Where Field” list box. With the *FieldName* selected, the “is...” list box will become enabled, allowing the End-user to specify the logic operator that will be used against the selected *FieldName* from the selected File. With the operator selected, the value to be used with the selected operator against the selected *FieldName* can be chosen. The choices of values can either be another data source [Field](#), a [Variable](#), a [Constant](#) or a [List](#). For a sample of a basic Selection Logic statement, see [How To Use Selection Logic](#).

With one selection statement entered, the End-user can logically AND this statement to a new statement by clicking the “And” button. An example of this might be the need for: all the EMPLOYEES named SMITH and all PERSONNEL ID’s less than 500000. The more definitive a query’s Selection Logic, the more concise the resulting data. For more information on adding logic during the query building process, please see [And / Or Logic](#) in the glossary section of this document, the Help section of NatQuery, for graphical instructions please visit our web page; [http://www.networks-inc.com/Products/Slide\\_Show/slide\\_show.html](http://www.networks-inc.com/Products/Slide_Show/slide_show.html) and click on the link entitled “Building an End-User Extraction”.

# NatQuery Users Guide

## Modifying/Deleting Selection Logic

Selecting the “Selection Logic” statement to be modified from the “Current Selection Logic” list box the End-user may interact with the graphical elements in the “Selection Logic To Apply” frame to apply the desired changes.

To delete the entire Selection Logic, click the “Clear All” button. To delete a single line of Selection Logic, highlight that line and click the Clear button.

To append Selection Logic in an existing query the user would begin by highlighting the statement in question and clicking the Append button, and proceed as outlined in the section [Entering Selection Logic](#).

## Notes to the User

If a constant was defined but not used in the query, the variable for this constant definition is not deleted automatically when the constant is removed from the selection logic. This variable may be deleted by the End-user through the [List Query Variable](#) window. A variable being used in selection logic may not be deleted from the List Query Variable list without deleting the entire query.

To cancel out of Selection Logic at any time, click Cancel to close the window. If the user was in the midst of adding logic to an existing statement, as in an [Append](#) situation, then the original statement will be retained but the new logic will be deleted when the user clicks Cancel. Be aware that if this is a new query and the user is entering selection logic, clicking the cancel button will clear all the entered selection logic and the user will be returned to their query on the open desktop. In order to begin entering selection logic for their query the user will need to click the Modify button at the bottom of the Query Selection Logic window.

## Advanced Query Save

The Advanced Query Save function allows the user to save any changes they have made to the default values in the selection logic for both Send to Server and List Program Options windows. Please note that the two windows are independent of each other and the user may use/save different options for each window. In this way, a user’s requirements for a specific query are preserved for later use while not disrupting the administratively set default values they will/might use in their next query. Be aware that this action is the equivalent of a Save function; the Save operation will overwrite the original query – the Save As operation creates a new query and will require a new program name.

## Building a Dynamic Variable

See this and other graphical presentations on the NatWorks Web Site @ [http://www.natworks-inc.com/Products/Slide\\_Show/Building\\_a\\_Dynamic\\_Variable.pps](http://www.natworks-inc.com/Products/Slide_Show/Building_a_Dynamic_Variable.pps)

## Building an End-User Extraction

See this and other graphical presentations on the NatWorks Web Site @ [http://www.natworks-inc.com/Products/Slide\\_Show/Building\\_an\\_End-User\\_Extraction.pps](http://www.natworks-inc.com/Products/Slide_Show/Building_an_End-User_Extraction.pps)

# NatQuery Users Guide

## Building a Datawarehouse – ETL Integration

See this and other graphical presentations on the NatWorks Web Site @ [http://www.natworks-inc.com/Products/Slide\\_Show/ETL\\_Integration.pps](http://www.natworks-inc.com/Products/Slide_Show/ETL_Integration.pps)

Periodically NatWorks will update or add slide shows for client edification. You may find additional slide shows on our web sit @ [http://www.natworks-inc.com/Products/Slide\\_Show/slide\\_show.html](http://www.natworks-inc.com/Products/Slide_Show/slide_show.html)

## Solving an Inventory/Sales Date Range Problem

Inventory / Sales problem (Backorders, Outstanding orders, current orders):

What is needed to return open orders within the date range of 5 days before today's date (current date) and 26 days past today's date?

Open a New Query and select a File for use such as Sales and select a field such as Open Orders.

You will need 3 variables. Here are the steps you need to set these up for use in Selection Logic:

The first variable will hold the current system date:

1. Create a new variable. Assuming the new query is already opened on the NatQuery desktop, use icon six [Query Variables](#); this opens the Define Query Variable window. Click the Add button; give the new variable a name (we'll assume it is to be called CURRENT-DATE).
2. This will be a [Dynamic Date](#) variable, so select Date from the Format combo box.
3. Click on the Dynamic radio button and then click the Define Value button - this brings up the Define Dynamic window.
3. Click on the Dynamic Date button - this will populate the combo box with CCYYMMDD. This is the only format available, so click the OK button on this window to return to the Define Query Variable window.
4. Click OK on the Define Query Variable window. Current-Date now equals CCYYMMDD.

The second variable will be the Current Date - 5 days:

1. Create a new variable using the Add button in the Define Query Variable window. Name it appropriately (we'll assume it is to be called DATE-MINUS-5).
2. Select Date from the Format combo box (on the Define Query Variable window).
3. Click on the Expression radio button, and then click Define Value. This will open the [Define Expression](#) window.
4. Click on the Variable radio button. A combo box containing the currently defined variables will become visible. Select #CURRENT-DATE (defined above).

# NatQuery Users Guide

5. Click on the Minus (-) command button.
6. Click the Value radio button; enter 5 in the text box that appears, and click on the Enter command button.
7. Click OK to return to the Define Query variable window. DATE-MINUS-5 now equals Current-Date minus 5 days.

The third variable is the same as the second, but you will add 26 instead of subtracting 5 (We'll assume it is to be called DATE-PLUS-26). Follow the instructions above entering a + sign instead of a – sign and use 26 instead of 5 in the text box.

8. Click OK to return to the List Query Variables window. DATE-PLUS-26 now equals Current-Date plus 26 days.

Clicking OK on the List Query Variable window brings the user back to the Selection Logic window or to the open query on the users desktop and they will need to click the Modify button on the Query Selection Logic window to continue.

In the Where Field of the Selection Logic window create two separate conditions, one for: (Where Field) #Current Date, (is) >, (than) (Variables) #DATE-MINUS-5, and a similar expression for #Current Date < #DATE-PLUS-26.

In the Current Selection Logic window it would look like:

```
DYNAMIC-CCYYMMDD > CURRENT-DATA-5  
And  
DYNAMIC-CCYYMMDD < CURRENT-DATA+26
```

Upon submission NatQuery will prompt you to enter the Dynamic dates as needed.

## End User Windows

Please note that various windows shown here may differ slightly from those seen in your version of NatQuery depending on the server environment that ADABAS and Natural are located on.

### Select File

Description (from the “Help” directory of NatQuery)  
See also the first icon, “[New Query](#)”.

The Select File function allows the End-user to select the files from which data fields can be used for extraction. This window will display all files that are available to the user for their data extraction needs. After the first file is selected, the files that are available for selection will be limited to just those files that have a valid [relationship](#) to the currently selected files. The user chosen Files will appear in the “Query Selected Files” window.

# NatQuery Users Guide

## How to Select a File

To select a file for extraction, the End-user may highlight a specific filename and then click the Select button, or they can double click the *FileName*.

After a file has been selected, the End-user will be presented with the Select Fields from *FileName* window, which will allow the required extraction fields to be designated. The previously chosen “Query Selected Files” will appear in a window behind the Select Fields from *FileName* but will be disabled until the User takes some action within the Select Fields from *FileName* window.

## (Window) Query Selected Files

Description (from the “Help” directory of NatQuery)

The Query Selected Files function is a [child window](#) of the NatQuery desktop, and will be displayed anytime a Query is on the NatQuery desktop and this Query has had at least one file selected for extract processing. This window allows for immediate access to the central element of Query generation – the designation and handling of extract files. This window will display all files that are designated to have data fields extracted by the Query.

## How to Use the Query Selected Files Window

If the End-user desires to add a new file to the currently open Query, then this can be accomplished by clicking on the Add button. This will open the Select File window. A query is restricted to a maximum of five primary files with these files all having a common relationship element as designated by the Administrator.

If the End-user desires to modify the current field extraction information for a selected file, then this can be accomplished by selecting the file and then clicking on the Modify button. This will open the Select Fields from *FileName* window.

As files are selected for extraction processing, these files will be assigned a unique file prefix, and will then be displayed in the list box of the Query Selected File window.

If the End-user desires to delete a file from the current Query, then this can be accomplished in the Query Selected Files window by clicking on it and then clicking on the Delete button. NatQuery will then respond with a confirmation for the deletion request. Please note that a file can be deleted from a Query only if there are no references to the file in any variable or any Selection Logic statement. If the Selection Logic is empty and the only file in the Query is deleted from that Query, then NatQuery will respond as if the End-user requests to open a new Query, and will clear the desktop and present the Select File window.

## Select Fields From *FileName*

Description (from the “Help” directory of NatQuery)

The Select Fields from *FileName* function allows the End-user to select data fields for extraction from a specified file. On the left side of this window, all available fields from the specified file are presented. On the right side of this window, the designated fields to be extracted are presented (if any currently exist). This function is accessed from the Select Files window or the Query Selected Files window.

When looking at the Select Fields from *FileName* window, the user may see fields which have a double “#” character or another double set of characters these marks designate a variable. If these types of fields are

# NatQuery Users Guide

displayed, these represent administratively “[Pre-Defined](#)” variables that have been made available to users so that these Users do not need to define these variables repetitively for each query. An example of a predefined variable might be a time/date shown in a “company” excepted format or an expression variable made to summarize employees’ salary occurrences.

See also [Concatenate/Flatten Option](#), [Occurs](#), [Recurring Field Handling](#) and [Define Lookup Retrieval](#)

## To Select Fields for extraction

To select a field for extraction, the End-user can either select a field and then click the Add button, or can double click the field. If all possible fields from the file are desired for extraction, click the Add All button.

When using the Select Fields from *FileName* function, you may see that some of the fields that are listed for selection are followed by text enclosed in parenthesis, or that some fields are preceded by a ">" sign. Fields that are followed by text in parenthesis are known as recurring fields, for additional information on using recurring fields; please refer to [Recurring Field Handling](#). Fields which are preceded by a ">" sign are called lookup fields. Refer to [Define Lookup Retrieval](#) for more information.

When recurring fields are selected for extraction, the default occurrence information as specified by the NatQuery Administrator is shown in parenthesis following the field. Once a recurring field has been selected, the default occurrence information for that field may be altered through the [Occurs](#) button. When the Occurs button is pressed, either the [Specify Field Occurrences](#) or the [Specify Table Occurrences](#) windows will be displayed, depending on what the type of recurring field is.

When selecting [Pre-Defined variables](#), the End-User may notice that the selection of one such pre-defined variable causes one or more other pre-defined variables to become part of the selected fields. This is normal, and is caused by NatQuery parsing the definition of the selected variable and determining that the variable has dependencies on other pre-defined variables – and therefore auto selects these dependent variables.

If the user has not yet chosen a file, only the “Cancel” and the “Help” buttons will be enabled. The User is only allowed to choose one [primary file](#) at a time but up to five primary files may be chosen altogether for use in a single query. The User may highlight a given File and then click the “Select” button or double click the chosen File and proceed to the “[Select Fields from FileName](#)” function. If the User has chosen all the Primary Files they wish to use then they should click OK and then the “Continue” button when the “Select File” window next appears.

## Deleting Fields from a Query

Fields cannot be deleted directly from the [Query Selected Fields](#) window. In order to delete a field from a query the user will need to highlight the parent File found in the [Query Selected Files](#) window and then click Modify. This will bring up the Selected Field from *FileName* window; from here, the individual field(s) can be removed by selecting the field in the right window and clicking the Delete button, or by double clicking on the field. To delete all currently designated extraction fields, click the Del. All button. If the selected File contains a [pre-defined variable](#) and this variable is unwanted, it will need to be deleted through the Query Variables window, [icon six](#). If a File contains pre-defined variables, the user must designate at least one non-pre-defined variable for use in their query before NatQuery will allow extraction processing to take place.

# NatQuery Users Guide

A situation where it may be necessary for the user to delete a field is the situation where the user has chosen two or more files and these files have a relationship such that they share certain fields such as ISN or Personnel ID. These fields might look like this; File01.Personnel ID and File02.Personnel ID and either the File01 or the File02 field will need to be deleted before NatQuery will execute the query.

## Search function

In some instances, the File chosen will have a lengthy list of available Fields. The user may wish to utilize the “Search” function of the “Select Fields from FileName”. This button brings up the “Search for Field”.

The Search for Field function is used to quickly find fields that may exist within a file, allowing a user to search for fields that begin with, contain, or end with a specified string. The Search for Field function can be called from several locations within NatQuery.

## How to Use the Search for Field function

To use the Search for Field function, the user will first decide on how they wish to search for the field. By default, the Search for Field function will assume that the user wishes to find fields that contain a given string value. Alternatively, the user may elect to find fields that either begin or end with a given string value. The user will therefore start by selecting which search option they wish to use by clicking on the appropriate radio button, with the options being:

- Begins with Search String
- Contains Search String
- Ends with Search String

After selecting the appropriate search option, the user will enter the criteria (string) to be used for the search, through the blank text field labeled, Search String.

When initially presented, the list box at the lower portion of the screen labeled, Fields that Match Search Criteria will display all fields that will be searched. As the user enters text into the Search String field, the contents of the Fields that Match Search Criteria will be dynamically changed to reflect the entered search value. Because more than one field may meet the search criteria the user must highlight any fields that are desired in the Fields that Match Search Criteria window before clicking the OK button. This action will populate the Select Fields To Retrieve window with the highlighted field names.

If the user has all the necessary Fields they can now click OK on the Select Field from *Filename* window. More fields may be added directly from the Possible Fields to Select or if using more than one File the user may wish to add Fields from these Files using the method described here...

To cancel this function at any time, click Cancel to close the window, and the End-user will be returned to the Select Files window.

## Occurs function

The “occurs” function can be found on the “Selected Fields To Retrieve” side of the “Select Field from FileName” window. Clicking the Occurs button brings up the Specify Field Occurrences window.

# NatQuery Users Guide

The Specify Field Occurrences function allows the End-user to specify the occurrences of a [recurring field](#), which represents an [array](#). For the specified field, the End-user can change the dimensions of the array extracted within the maximums designated by the NatQuery Administrator. When this window is displayed, the field being handled is either a Multi-Value ([MU](#)) field or an Element field of a Periodic Group ([PE](#)). This window will be presented to the End-user when the End-user requests to extract a recurring field.

## How to Specify Field Occurrences

When specifying Field Occurrences, it should be remembered that the field has only one dimension. The Options selection box provided on the lower left of the window controls the "width" of the field occurrences, or array. The options available for the width of a field include All, Range and Highest.

- If All is selected, the field width will be all occurrences of the field up to the administratively controlled maximum.
- If Range is specified; a range of occurrences within the administratively controlled maximum will designate the field width.
- The Highest option will allow for the highest n occurrences within the administratively controlled maximum.

At any time, the End-user can return to the default values for this field, as determined for this field by the NatQuery Administrator, by selecting the Default checkbox.

When the desired field dimensions have been selected, the End-user can click the OK button to close the window.

To cancel this function at any time, click Cancel to close the window and return to the field's original values.

Please Note; the Specify Field Occurrences window accessed from the Select Field From *FileName* or Query Select Fields window differs from the window found when using a field in Selection Logic that requires occurrence information in that the user will need to close and reopen the latter to return to the default values of the field if they first change these default values and need to reset them.

Please Note; when using the highest option, it should be noted that the values returned would be the highest n values specified. If the number of occurrences, which actually exist for the field, is less than the specified n value, NatQuery will return n occurrences that will include the highest n occurrences.

Please Note; The row of boxes shown in this window is for visualizing an array structure and the dimensions of this row of boxes will not correspond to the selected dimension values for the field.

For general information on using recurring fields with a query, please refer to [Recurring Field Handling](#).

## Specifying Table Occurrences

The Specify Table Occurrence function allows the End-user to specify the occurrence information to be returned as a function of the specified Fields to be returned or to be used when defining an expression, which

# NatQuery Users Guide

references a recurring field, which has two dimensions. These two dimensions can be thought of as a Table with related data in both columns and rows. It should be noted that when defining an Expression variable that references any recurring field, the [Expression variable](#) requires that a specific unique occurrence be specified. This window will be presented to the End-user when a recurring field stored as a table is referenced in selection logic.

The field for which occurrence information is being modified will be displayed at the top of the window. On the left of the window, the Occurrence selection box is used to designate the appropriate row of the occurrence from the "depth" of the table. The occurrence specified here could be any occurrence up to the administratively controlled maximum for the field.

The Occurrence selection box in the middle right of the window is used to designate the appropriate column of the occurrence from the "width" of the table. The occurrence specified here could be any occurrence up to the administratively controlled maximum for the field.

The intersection of the selected occurrence row and the selected occurrence column will be the specific occurrence that will be used in the Expression variable.

When the required occurrence information is specified to the users' satisfaction, click the OK button to close the window.

To cancel this function at any time, click Cancel to close the window. If this occurs, the previously defined occurrence information for the field will be retained.

For general information on using recurring fields with a query, please refer to [Recurring Field Handling](#).

## Query Selected Field

The Query Selected Fields window is a [child window](#) of the NatQuery desktop, and will be displayed anytime a Query is open on the NatQuery desktop and this Query has one or more fields or variables selected for extraction processing. This window will display all fields and variables that are to be extracted by the Query.

### How to Use the Query Selected Fields Window

The fields and variables selected for the extraction process will be displayed in the list box of the Query Selected Fields window.

If the End-user desires to either

1. Add additional fields from extraction or
2. Delete currently selected fields from extraction processing,

These actions can be taken in the Query Selected Files window by first highlighting the File containing the Fields that need modification and then by clicking the Modify button.

# NatQuery Users Guide

## Re-sequencing *FieldName* order

The Query Selected Fields window supports the [re-sequencing](#) of the fields and variables that will be included in the data extract. This is accomplished by highlighting (clicking on) a field, then using the “Up” or “Down” buttons to change the order of the fields or variables within this window. This changes the order in which they will appear in the extract.

The Query Selected Fields window also supports the ability to manipulate occurrence information for any recurring field selected for extraction. Recurring fields can be easily identified, as they will have a field name followed by the current occurrence information specified for the field enclosed in parenthesis. To modify occurrence information for a recurring field, select the field and then click on the Occurs button.

The Query Selected Fields window also supports the ability to view information for any given field, if the information has been retrieved from Predict, or entered manually. To view PREDICT information for a field, select the field and then click on the Predict Info button. If no information is available, then the PREDICT Information window will appear with a message to that effect.

Depending on the type of recurring field selected, NatQuery will then present either the Specify Field Occurrences window or the Specify Table Occurrences window. These windows allow the End-user to specify the occurrence information that is required for the given recurring field.

For general information on using recurring fields with a query, please refer to [Recurring Field Handling](#).

## Selection Logic

See also, [Building a Simple Query](#).

## Defining a Constant Variable

The Define Constant *variablename* function allows the End-user to create a constant variable. A Constant variable is an End-user-defined variable that has an assigned value that remains constant. This function can be opened from either the Selection Logic window or the Define Variable window.

The Input Value for *variablename* field allows the End-user to enter a constant value. If the format of the requested constant variable was alphanumeric, then by utilizing the Upper Case Only checkbox, the End-user can control the upper or lower case value of the field as it is entered. If the format of the requested constant variable was numeric, then the Upper Case Only checkbox will not be visible.

The user should type in the desired value of the Constant field into the Input Value for *variablename* field. When done, click the OK button to close the window and define the Constant variable.

At any time, the user may cancel the Define Constant Variable window by clicking the Cancel button. By not clicking the OK button, any unsaved Constant Variable creations or modifications will be lost, and this window will close.

Note: When the Define Constant *variablename* function is accessed directly from the Selection Logic window, this has the affect of creating a [variable](#) type of constant.

# NatQuery Users Guide

If a constant was defined but ultimately not used for some reason, the variable for the constant definition is not deleted automatically when the constant is removed from the selection logic. If necessary, this variable will need to be deleted by the End-user. Please see the topic entitled [How to Delete a Variable](#) for help with this.

For general information regarding how to use variables with a query, please refer to [Variable Handling](#). For information on creating a variable, please see [Building a Dynamic Variable](#).

## Define List Variable

The Define List Variable window allows a user to enter / modify a unique NatQuery variable called a List variable.

When entering Selection Logic into a Query, the user is limited to only using “and” statements when specifying multiple selection criteria. While this approach is meant to simplify the entry of Query specifications, it does impose a barrier when “Or” logic is required. The List Variable has been designed to minimize this limitation. Using a List Variable allows NatQuery to process selection logic with “implied or” logic.

If the user wants to select records from an Employee file that have a specific last name and who are in a specific city, then normal “and” processing would look like this:

```
FILE01.LAST-NAME = SMITH  
AND  
FILE01.CITY = BOSTON
```

However if the user wants to select records from an Employee file that have a specific last name and who are located in one of several cities, then normal “and” processing would force the user to create multiple queries to retrieve all cities that the last name is found in. It is exactly in this instance where a List Variable may be utilized, as in:

```
FILE01.LAST-NAME = SMITH  
AND  
FILE01.CITY = (“NORTHFIELD” or “BOSTON” or “BURLINGTON”)  
                  >User created ↑ List Variable<
```

## How to use the Define List Variable window

At the top of this window, the selected field, its format, and length will be represented. The text box directly below allows the user to type in a new value to add to the list. Once a value is typed in, the user can click the Add button to physically add this value to the list of values displayed in the Current List Items text box. Once the Add button is clicked, the text box is cleared in preparation for the next item to be entered.

To modify an existing list item, first highlight the list item to select it, and then click the Remove button, then add the correct value into the uppermost text box, and then click Add.

To remove an existing list item, first highlight the list item to select it, and then click the Remove button.

# NatQuery Users Guide

Once the List Variable contains all the desired items in the Current List Items text box, click the OK button to save the list. Clicking the Cancel button at any time will delete the users' choices and exit the user from this function.

## Exiting the Define List Variable window

Once the elements have been added to the List Variable frame, clicking the OK button will internally create or modify the List Variable and will then close the Define List Variable window.

At any time, the user may cancel the Define List Variable window by clicking the Cancel button. By clicking the Cancel button, any unsaved List Variable creation or modification will be lost, and the window will close.

## Defining a Field Value

The File/Field Value function allows an End-user to specify a Selection Logic statement that has, as its last operand, a reference to a file field, instead of a reference to a variable or a constant. In the Available Files selection box, lists of the files that are currently defined to the Query are displayed. If only one file has been defined to the Query, then the Available Files selection box will display this file, and the selection box will be disabled. If more than one file has been defined to the Query, then the selection box will be enabled and will allow you to specify the file from which the specific field of interest resides.

The Available Fields for File selection box will allow for the selection of any field from the currently selected file in the Available Files selection box.

Select the file of interest and then select the specific field from that file to be used in the Selection Logic statement. When the appropriate field is selected, the End-user can click the OK button to close this window and return the selected file/field to the current Selection Logic statement being edited in the Selection Logic window.

To cancel this function at any time, click Cancel to close the window. If this occurs, any previously existing field value will be retained. If there is no pre-existing value or new value information, then the information will be lost by clicking the Cancel button.

## Define Variable - Variable Handling

The fourth variable found in the Selection Logic window, marked "Variables", allows the user five options to build variables for their query. Clicking the Variable button will bring up the the Define Query Variable window, this window can also be accessed through the [Query Variable](#) icon.

The Define Query Variable function allows an End-user to either enter or modify the base information required to define a variable, and then to further classify the variable as a type: Redefinition, Constant, Expression or Dynamic. The Define Variable function allows for the definition of new variables to a Query, as well as allowing for the definition of existing variables to be modified.

# NatQuery Users Guide

## How to Define a New Variable

The first window the user will encounter after clicking the Variable button is the List Query Variables window. For more information on modifying existing variable or creating a new variable please refer to the section in this document entitled; [The Sixth Icon “Query Variable”](#).

When the Define Query Variable window is first opened for a new variable, the field Variable Name will be initialized to a unique variable name in the format of "Variable-xx", where "xx" is a unique number. Every variable must have a unique name within a query, and to optimize the use of variables within a Query, it is recommended that you supply a name for the variable that will allow for easy identification later. Variable names are limited to a length of 30 characters.

Through the Define Query Variable window the user can create a variable or variables that can then be used later as a “Where field”.

## [Include with Extracted Fields](#)

The Include with Extracted Fields checkbox allows the user to include the new variable as a field to be extracted by the Query. In such targets as Excel, the new *VariableName* will become a column header along with the other Query Selected Fields. This variable will not show up on the [Selecting Fields from FileName](#) window.

The [format](#) and length of the variable can be defined or sometimes modified. The Format and Length frame offers the user seven format choices: Alpha Numeric, Binary, Date, Integer, Numeric, Packed Numeric, and Time, the length options will vary depending on the format choice.

## Define Redefinition variablename

The Define Redefinition variablename function allows the End-user to create a redefinition variable. A redefinition variable is a variable that represents a "piece" of a data field. As an example, the End-user may want to extract just the first five characters of a 20-character file field, or they may want to only return the area code (first three numbers) of a phone number. A redefinition variable could be used to provide this data.

## Redefinition Variable

The File Containing Field to Redefine selection box is used to designate the file that contains that data field for which a redefinition is required. If there has been only one file defined to the current Query, then File Containing Field to Redefine displays this file name, and the selection box will be disabled. If more than one file has been defined to the current Query, then the selection box will be used to select the file that contains the data field for which a redefinition is required.

With a file selected in the File Containing Field to Redefine field, the Available Fields from File selection box will become enabled. This will allow the End-user to select the base data field for which a redefinition is required. Once a field is selected, the From selection value will be initially set to the first character in the data field, and the To selection box will be set to the last character of the data field. Using the From and To selection boxes, the user may specify the redefinition value of the data field required.

When the correct values are entered for all selection boxes, clicking the OK button will close the window.

# NatQuery Users Guide

To cancel this function at any time, click Cancel to close the window. If this occurs, any previously existing redefinition will be retained while the work in progress will be deleted.

Please note: For general information regarding how to use variables with a query, please refer to [Variable Handling](#).

See [Define Constant variablename](#) for more information.

## Define Dynamic variablename

The Define Dynamic Variablename function allows the End-user to define a Dynamic variable. A Dynamic variable can be either date related or can be based on input from the End-user.

### How to Define a Dynamic Variable

If a Dynamic Date variable is desired, the End-user will click the Dynamic Date button, then the Define Value button; this will open a selection window. In this window, the End-user may then select a date variant based on the most commonly used combinations of century, year, month, and day. A "CC" represents a two-character century. A "YY" represents a two-character year. A "MM" represents a two-character month. A "DD" represents a two-character day.

When a Dynamic Date variant is used, this variant will be properly set to the appropriate value based on the current system date of the workstation.

If a Dynamic Input variable is desired, the End-user will click the Dynamic Input button, which will present selection boxes for the Format and Length of the input variable.

When a Dynamic Input variable is used, the End-user will be prompted for the current desired value of the input variable, which will then be used for query generation.

When the desired dynamic variable has been defined, click OK button to close the window.

To cancel this function at any time, click Cancel to close the window.

Please note: For general information on using variables with a query, please refer to [Variable Handling](#).

## Define Expression variablename

The Define Expression Variablename function allows the End-user to define an Expression Variable. An expression variable allows addition, subtraction, multiplication, and division of fields from one or more files that share a relationship ([cardinality](#)). Example; the user wishes to create a variable that has all the occurrences of an employee's salary added together such that the user can report on all Employees with a total salary above a certain amount.

An expression variable cannot be Alphanumeric.

Please note that it is a limitation in the current version of NatQuery that an expression variable once defined cannot be subsequently modified. To effect a modification in an existing expression variable that is currently defined, one must clear the existing elements of the expression variable and then re-enter them as desired. This

# NatQuery Users Guide

is accomplished through the Current Selection Logic frame of the Selection Logic window using the Clear or Clear All buttons.

## How to Define an Expression Variable

This function operates in a somewhat similar fashion to a calculator. To use this function, first select an appropriate operand, which may be either:

- A file/field combination
- An existing numeric variable, or
- A value

With an operand selected, the Operator fields will become enabled, allowing an operator to be supplied to your expression. Once an operator is entered, you will then select another operand to complete the expression. At this point, if the expression is complete, you may click the OK button to accept the expression variable and close the window. Otherwise, you may continue to add to the expression by supplying another operator followed by another operand. An expression may be up to 77 characters long.

## Conditionally Assigned

The [Conditionally Assigned](#) checkbox option determines whether this variable is conditionally assigned. Setting this option and clicking on the Define Condition button will present the Define Initialization Condition for Variablename window, where the condition for initializing this variable will be entered. Not all variables can be conditionally assigned. Only, Expression variables may have initialization conditions associated with them – Constant, Redefinition, Compress, and Dynamic variables may not.

The Conditionally Assigned checkbox option determines whether this variable is conditionally assigned. Setting this option and clicking on the Define Condition button will present the [Define Initialization Condition for Variablename](#) window, where the condition for initializing this variable will be entered.

Of the available variables, the Expression variable is the only variable that can be conditionally assigned by the user.

## Define Initialization Condition for Variablename window

The Define Initialization Condition for Variablename window is used to define the condition under which the variable *Variablename* will be initialized. If the variable is numeric, the default value will be zero. If the variable is alphanumeric, the default will be an empty string. Only when the specified condition is met will the variable set to the value specified in the appropriate window.

## To set an Initialization Condition

Select the first operand of the condition. To do this, click on either the Field From File or Variable option buttons. If Field From File is specified, then select the appropriate file and field using the drop-down list boxes in the Operand One frame. If Variable is selected, then choose the appropriate variable from the drop-down list box in the Operand One frame. The appropriate list box(s) will be displayed depending on which option is selected.

Second, select an operator from the list in the Operators frame.

# NatQuery Users Guide

The third step is to select whether the second operand will be a variable or a constant value. If there are no other variables defined, or none of the defined variables is a type that matches the first operand, then the Value option will automatically be selected. Otherwise, both the Value and Variable option buttons will be available. At this point, select the desired variable, or enter a value into the text box (visible only if the Value option has been selected). If the first operand is a Time variable, then you must enter either a time or date/time value for the second operand or specify another Time variable for the second operand. If Value has been selected (and the first operand is not a Time variable), then an Upper Case Only checkbox will become visible. This checkbox is checked by default. Checking or un-checking it will allow you to choose whether text is in all capital letters or not.

Once a condition is specified, clicking OK will exit this window and set up the condition. Clicking on the AND button will allow the entry of another Initialization Condition. Clicking Clear at any time will clear the condition(s). This is the only way to make changes to the condition logic.

Clicking Cancel at any time will exit this window without setting up the specified condition(s).

At any time, the Clear button may be used to clear the entire current expression so the end user can start over.

To cancel this function at any time, click Cancel to close the window. If this occurs, any previously defined expression value for the specific variable will be retained; however the work in progress will be deleted.

Please note: For general information regarding the Variable option when building a query, please refer to [Variable Handling](#).

## Define Compress variablename

The Define Compress Variablename function is used to create a Compress variable that contains a number of components as well as spacing options. These components may be fields from a file, other variables, or user-defined strings. The two spacing options are used to specify if spaces should be left between components and are designated as Space or No Space on the form. This is useful for combining multiple fields and/or variables, as well as user-defined text, into one extracted field. Example; creating a variable that adds a first name to the second name with a “space” in between such that the user might use this compressed variable to build a mailing address label.

## How to Create a Compress Variable

The Define [Compress](#) Variablename window contains several option buttons, which are labeled Field from File, Variable, and Text. These option buttons allow the user to switch between which type of component will be added to the Compress variable. There are also two command buttons labeled Space and No Space. These allow the user to specify whichever spacing option they require. A Compress variable is built by alternating between the selection of components and the selection of spacing options.

A Spacing option must be specified between each component. If a space is desired, click on the Space button. If no space is desired between the components, click on the No Space button.

To select a field from a file, click on the Field from File option button. Two drop-down list boxes will appear – one labeled File and one labeled Field. Select the desired file from the File drop-down (this drop-down list box

# NatQuery Users Guide

will be disabled if there is only one file specified in the query). Then select the desired field from that file from the Field drop-down list box. Once selected, the field will appear in the Selected Components list box.

To select a variable, click on the Variable option button then select the desired variable from the drop-down list box that appears. The variable will then automatically appear in the Selected Components list box.

To insert plain constant text, click on the Text option button. Enter the desired text into the Text to Insert text box then click the Add button. This will insert the text (in quotes) into the Selected Components list box. Components and spacing options may be cleared one at a time, starting from the end of the list by clicking on the Clear Last button below the Selected Components list box. To start over entirely, click on the Clear All button below the Selected Components list box.

To create the variable, click OK this will also exit the user from the window. The OK button will not be enabled unless at least one component has been selected. To exit at any point without creating the variable, click Cancel.

## How to Modify a Variable Definition

To modify the definition of an already existing variable, the user would begin by highlighting the variable they wish to change on the List Query Variable ([icon six](#)) window. This will enable the Modify button, clicking this button will open the Modify Query Variable window. The variable's name can be modified to any value that is unique for the Query. This window will initially contain the current definition of the variable of interest. The format and length of the variable can be modified, as well as the specific type of variable - [Redefinition](#), [Constant](#), [Expression](#), or [Dynamic](#). The user is also given the choice as to whether or not the variable is to be extracted with the Query's other extract fields in the form of a checkbox option. If the variable type is modified, the appropriate window will be presented to capture the specific information required for this new type.

When you have finished modifying the variable, clicking the OK button will close the window.

After following the appropriate variable definition instructions stated above, the user would click OK on the Define Query Variable window and then click OK on the resulting List Query Variable window. On the List Query Variable window the user will find their newly developed variable. This variable will also be listed in the "Where Field" column of the Selection Logic to Apply frame of the Selection Logic window.

For general information, including adding a new variable, deleting a variable, or how to use variables with a query, please refer to [Variable Handling](#) or the section entitled [Define Query Variable](#).

Note: When modifying a variable that is used in selection logic or in the definition of another variable; the name, format, and length cannot be changed.

## Deleting Selection Logic

Assuming the user wishes to delete a single line of existing selection logic they will need to highlight that line in the Current Selection Logic frame. The user can now click the Clear button found in the lower right side of the window. Please note in selection logic that uses the "And" operator to join multiple statements; the "And" statement will need to be highlighted and Cleared as well. A query may not begin, end, or have back to back "And" statements.

# NatQuery Users Guide

The Clear All button will delete all the existing selection logic found in the Current Selection Logic frame.

If the user saves the (previously saved) query after clearing or [modifying](#) the existing selection logic then any prior logic will be lost.

See [Modifying Selection Logic](#)

## **Please Note:**

A variable that is referenced in selection logic or in the definition of another variable used in an existing query cannot be deleted. For more information, please see [How to Delete a Variable](#).

For general information on variables, please refer to [Variable Handling](#).

## **Windows**

### **Window Concatenate / Flatten Fields**

In order to provide for the flexible extraction of recurring field structures, NatQuery allows recurring fields to either be "[flattened](#)" or "[concatenated](#)". This function allows the End-user to designate the type of action to be performed on each recurring field being handled for the extraction. Concatenation causes recurring fields from a record to become individual fields in a single extract record. Flattening will cause recurring fields from a record to be split into multiple extract records. Each extracted record will contain a single recurring occurrence and a complimentary indexing field.

See also [Recurring/Repeating Fields](#)

### **Window #Target Name Configuration Field Information**

This window will always be shown with a caption in the following form:

<xxxxx> Extract Configuration – Field Information

where, <xxxxx> represents a given extract type, such as DB2 Loader, MS Access, etc.

This window is used to specify field names for the target database or file type, as well as field formats and lengths of certain extract types. There are two ways to modify the NatQuery-generated defaults. The first is to select the field entry (in the Database Field Information frame), and then click on the Edit button. The second is to double-click on the entry that you wish to modify. This will present the Output Field Information window, which is used to modify information pertaining to the selected field.

To continue with the request (after making all necessary changes), click the OK button.

To cancel this window (and the current request), click Cancel.

# NatQuery Users Guide

## Window - ERROR message for Access Extract

Extract type: Download into Access

Microsoft's Access database has "reserved words". If a Field name is a reserved word then NatQuery will recognize this and prompt the user to edit the Field name. Reserved Words may differ dependent on the version of Access being used in your work place, using the Help feature of Access you may look up the list of reserved words that applies.

## Window - ERROR message for Any2XML Extract

Extract type: Any2XML Extract

"No Any2XML User ID has been specified in the current configuration.  
Error message continued: Please use the NatQuery Configuration function to enter this User ID.  
This extract request will now be cancelled."

Any2XML is a third party product that requires a valid license key to operate. Please contact your Administrator for more information on obtaining this product for use in your work place.

## Window – Manage Scheduled Tasks

Extract types: Download to XML, RDBMS Loading, Download into Access, Download into Excel, and Download to PC File

In the current version of NatQuery (4.3.0) Scheduled Tasks can not be used with Windows 98.  
In the current version of NatQuery (4.3.0) Scheduled Tasks can not be used with Summary Processing.

This window allows the user to schedule the submission of their query at a future time with four options: Once Only, Daily, Weekly, or Monthly. In all four options the user must indicate the time for the submission as being AM or PM. Once the settings are complete if the user chooses they may save this query and the [Advanced Query Save](#) feature will retain the scheduled submission criteria.

The submission can be a Once Only, that is the query will be scheduled to run only once at a specific date and time in the future, or the submission can be scheduled to run Daily at a specified time such as every 24 hours at 12:00 pm, every 48 hours at 8:00 pm, or every 72 hours at 6:00 am, etcetera..

For both the Daily and the Weekly options the user can specify the Interval of days (24 hours per interval) or weeks (168 hours per interval) between the scheduled submissions. The Weekly option allows the user to specify the day of the week to run the query.

The Monthly option allows the user two options for their scheduled query submissions. In the first instance they may choose a particular date and specify which month(s), example; schedule query for the 9<sup>th</sup> of June August, and September only. In the second instance the user can specify a day of the week as well as the month or months to run their query, example; schedule query for the second Thursday of every January, February, and June.

# NatQuery Users Guide

Scheduled Submissions may be accessed for Editing or Deletion through the NatQuery menu bar; Administer/Schedule Task. Editing or Deleting a scheduled submission in this manner will immediately affect the query in question.

If the user has opted to not delete a query that is/was scheduled as a Once Only submission from within the Check Server window for whatever reason, then the only way to delete this query, or edit it, is through the Administer/Schedule Task menu.

## **(Window) Send To Server Option – Retrieval Limit**

Extract type: All extract types

If the user clicks the button to indicate they wish to limit the number of records returned but fails to enter a number in the box provided, they will receive the following message.

“If you request to limit the retrieval limit, then a limit must be entered.”

## **(Window) Generate Program**

If the user closes the “Send To Server” window by clicking the Cancel button, the following message will appear, “Query Generation was cancelled by User action.”

This message is not the same as a “Generation Error” message. If the user encounters a Generation Error message when retrieving the results of a query they should save the query (not delete it) and contact their Administrator.

## **(Window) Define Query Variable**

See [Query Variables](#) section for more information.

The [Define Variable](#) function allows an End-user to either enter or modify the base information required to define a variable, and then to further classify the variable as a type of Redefinition, Constant, Expression or Dynamic. The Define Variable function allows for the definition of new variables to a Query, as well as allowing for the definition of existing variables to be modified.

### **How to Define a New Variable**

When this window is first opened for building a new variable, the field Variable Name will be initialized to a unique variable name in the format of "Variable-xxx", where "xxx" is a unique number. Every variable must have a unique name within a query, and to optimize the use of variables within a Query, it is recommended that you supply a name for the variable that will allow for easy identification later. Variable names are limited to a length of 30 characters.

The [Include with Extracted Fields](#) option determines whether the variable will be included with the fields that are to be extracted by the Query.

# NatQuery Users Guide

The next step in creating a variable is to specify the format and length of the variable being defined, using the Format and Length selection boxes.

The End-user may now select the appropriate type of variable being defined by using the appropriate radio button in the variable type's frame, with the choices being [Redefinition](#), [Constant](#), [Expression](#), [Dynamic](#), or [Compress](#). Once the appropriate radio button has been chosen, then clicking on the Define button will open the appropriate window to handle the designated definition. Depending upon the choice of variable type, one of the following windows will be presented to the user:

- Define Redefinition variablename
- Define Constant variablename
- Define Dynamic variablename
- Define Expression variablename
- Define Compress variablename

Once the processing of the subsequent window is completed, the End-user will then be returned to the Define Variable window.

When the definition of the variable is complete, clicking the OK button will close the window.

To cancel this function at any time, click Cancel to close the window. This action will cause any unsaved information to be lost and cause the definition to revert to its previous setting or to the Define Variable window.

## How to Modify a Variable Definition

When modifying the definition of an already existing variable, this window will be initialized to contain the current definition of the variable of interest. This information is shown in the Current Definition of Variable frame. The variable's name can be modified to any value that is unique for the Query. The format and length of the variable can be modified, as well as the specific type of variable, [Redefinition](#), [Constant](#), [Expression](#), or [Dynamic](#). If the variable type is modified, the appropriate window will be presented to capture the specific information required for the new type.

Please note: When modifying a variable that is used either in selection logic, or in the definition of another variable, the name, format, and length cannot be changed.

When you have finished modifying the variable, clicking the OK button will close the window.

To cancel this function at any time, click Cancel to close the window. Upon cancellation, any previously existing definition of the variable will be retained, if this is a new variable the user will be returned to the Define Variable window.

For general information regarding how to use variables with a query, please refer to [Variable Handling](#).

# NatQuery Users Guide

## Window - Enter Dynamic Value for #Name

The Enter Dynamic Value for #variablename window is presented to an End-user when; The End-user has requested to process the query through the [Send to Server](#) or [List](#) functions.

When the Enter Dynamic Value for #variablename window is presented, NatQuery is about to begin the process of generation of the Natural code required to resolve the query specification. This means that all variables must be resolved.

### Entering a Dynamic Value

If the format of the requested dynamic variable was alphanumeric, then by utilizing the Upper Case Only checkbox, the End-user can control the upper or lower case value of the field as it is entered. If the format of the requested dynamic variable was numeric, then the Upper Case Only checkbox will not be visible.

Type the desired value of the dynamic field into the Input Value field. When done, click the OK button to close the window.

To cancel this function at any time, click Cancel. This action will bring up a window with the following message: Query generation was cancelled by user action.

Note: For general information regarding how to use variables, please refer to [Variable Handling](#).

## Error Window

### Extract Type: Extract to XML on Server

Build JCL/Script

Error Message;

The specified [JCL/Script](#) template:

Production XMLServer Process was not found for the specified server type (NT)

Send To Server request has been cancelled; please contact the NatQuery Administrator

This message means that the proper [JCL](#) or [Script](#) template has not been introduced to allow for the creation of an XML file on the server. Contact you Administrator for more information.

Please note this is an error message that might be found where ADABAS is installed on server type NT your error message may be specific to your installation [platform](#).

### (Window) NatQuery Performance Check

#### Extract Type: [Read All Records](#)

Message;

Warning!!! The query you are generating will read all records from file: #FileName

# NatQuery Users Guide

This type of extract has potentially negative performance implications, and unless this action is intended, usually means that no “key” fields are being referenced in the entered selection logic. Do you wish to continue to execute this query?

Click Yes to continue with the query if you are sure you want to submit this query. Click No to cancel this query and review the Entered Selection Logic.

The “negative” impact may be that 1) you will get all the records for the File and you do not need all the records and or 2) It may be expensive for the computer time involved in reading, writing, and returning all the records from a given File if it is a large file.

For more information regarding the implications of Read All Records, please contact your Administrator.

## **(Window) Send Request to Server – Are You Sure?**

### **Extract Type: All Extracts**

Message;

The request is ready to be sent to the remote server.

Are you sure you wish to send this request for processing?

Clicking on the “Yes” button will send your request to the server for processing. Clicking “No” will halt the processing of the query allowing the user to exercise other options on how they wish to proceed.

## **(Window) FTP Process**

### **Extract Type: All Extracts**

Message;

[Request number](#) # has been successfully sent to the server, using [Active Request](#) Slot #.

This is a verification message that lets the user know their Standard Request submission has been sent and that when they look for the returned data the results can be identified (if not by the program name itself) the Active Request or Slot number.

The User should note that they must click icon nine “[Check Server](#)” to monitor and retrieve their desired data.

## **Glossary**

### **Access**

Access – see [extract types](#) Microsoft’s Access™ is a PC based database.

### **ADD**

ADD - this button shows up on several windows including the [Query Selected Fields](#) window and the [List Query Variables](#) window. In the case of using, add, from the Query Selected Fields window, the user will be allowed to access the “Primary Files” for use in the required data extraction. Clicking the, add, button from the List Query Variables window allows the user to build a new variable that can then be used in the query.

### **All Records**

The All Records option is found in the Data Retrieval Options frame of the Send To Server window. The Data Retrieval Options frame contains two radio buttons: All Records and Intersection Records. It should be noted

# NatQuery Users Guide

that these options are available only when a query contains more than one DDM – with only one [DDM](#) in a query, NatQuery will default to All Records. For more information on the Data Retrieval Option frame, refer to the description of [Intersect](#) in this glossary.

To see an example please go to [Data Retrieval Options](#) contained in the Send To Server portion of this document.

Not to be confused with [Read All Records](#), that is a Selection Logic option.

## And / Or Logic

And / Or Logic – using the List Variable option a user has the capability to build a query containing “implied or” logic. Ex: Find employees in Boston and/or New York. See the description for [List Variable](#).

## Array

An Array is an orderly set of values given in a single dimension. Please note that this definition is intended as a visual aid to the user when working with [Occurrence](#) information. For the purpose of this document a multi-dimensional array will be called a table – see [Table](#) in this glossary.

## Append

Append is a term that means “in addition”. After using selection logic to build a query the user may decide this query will not provide the exact data they are looking for. To change or add to the selection logic in the existing query they will use the “[And](#)” button found in the “Selection Logic” window or by highlighting an existing line of selection logic, which allows the append button to become enabled.

## Cardinality

Cardinality - is a term that describes how one file can be related to another file. For NatQuery, there are 17 different types of cardinality supported; 16 types which are standard, and an additional cardinality type of “Lookup”. For the 16 standard types, cardinality is represented by an expression, for example 1:1 (One to One); that means that for every record in the first file, there is always 1 (and only 1) record in the related file. The 16 elements of cardinality are as follows:

Cardinality Element	Cardinality Element Meaning
C	None or One
CN	None or One or More
CM	None or One or More
N	One or More
M	One or More
1	One

## Child Window

A child window is a window that appears because of action taken in the subsequent (original) window. There are many such child windows in NatQuery that the user will encounter as they begin to define the query and how the resulting data will be returned. The most obvious example would be when the user wishes to open a new query by clicking the First Icon. This action opens the Select File window where the user will need to select the File which they will need for their query data, selecting a File will immediately produce the window, Select Fields from FileName. Select Fields from FileName is a child window of the Select File window.

## Compress Variables

[Compress Variables](#) - allow multiple components, such as fields from files, variables, and user-defined text to be placed together into one variable, usually for formatting data in a particular fashion. Each component of a compress variable must be separated by a <Space> or <No Space> tag. This is due to the way that NatQuery needs to build the compress variable statements in Natural. Text is surrounded by quotation marks (“) automatically, as this is also necessary for the building of the Natural statements.

# NatQuery Users Guide

For example, if a user wanted a single field that displayed a surname, followed by a comma (no space between surname and comma), followed by a space and then a first name, the following components would be selected in the Define Compress #variablename window:

```
FILE01.NAME  
<No Space>  
“ ”  
,  
<Space>  
FILE01.FIRST-NAME
```

To do this, start by selecting the NAME field from FILE01, using the drop-down list-boxes at the top of the window. Then click on the No Space button. Select the Text option button, and enter a comma (,) into the textbox that appears. Click on the Insert button to add the comma to the compress variable definition. Please note that you do not need to include the quotation marks – NatQuery does this automatically for you. Click on the Space button, and then select the Field from File option button. Select the FIRST-NAME field from FILE01.

The output of this compress variable would look something like this:

```
DOE, JOHN  
DOE, JANE  
:  
Etc.
```

## **Conditionally Assigned**

Expression variables may also be conditionally assigned, meaning that a user may specify a condition under which the expression variable will be initialized. For more information on this, please see the section; [Define Initialization Condition for Variablename](#), or use the Help documentation accessed through NatQuery toolbar.

## **Concatenate**

### [Concatenate](#)

Concatenate - instructs NatQuery to take each element of a recurring field and treat them as individual and separate fields in a single extract record. Concatenation of a recurring field means that for a single record being read from a data source file, a single record will be written to the physical extract file with separate fields for each element of a recurring field. The Concatenate option is found in the [Recurring Field Handling Options](#) frame of the [Send To Server](#) window.

See also [Flattening](#), [Recurring Fields](#)

### **Concatenate/Flatten Option**

Concatenate/Flatten Option – this frame can be found in the [List Program](#) window and in the [Send To Server](#) window. See also; [Concatenate](#), [Flatten](#), and [Recurring Fields](#) in this glossary.

## **Constant Variables**

[Constant Variables](#) - allows your Query to be supplied with a variable whose value never changes. For example, you may want to extract records from a file in which there is a field called CITY. As part of your selection logic, you would like to have the criteria that the only records that should be extracted from this file are records that have the value of "BOSTON" for CITY. Setting this criterion is possible while building your Selection Logic by filling in the Constant Variable window with the value of "BOSTON". The expression, File01.City = "Boston", then becomes a part of the Query Specification, with the value of this constant variable being used for the process of physically extracting only the requested records that include City equal to Boston.

# NatQuery Users Guide

## DDM

DDM - is an acronym that stands for Data Definition Module. A Data Definition Module is a description, or “picture”, of a file that exists on a data source. This description records vital information about the file in terms of what fields exist, the field’s format and length, the type of field, as well as “key fields” that provide a path into the file without having to read all records in the file.

## Datawarehousing Extract Type (DWH)

DWH Extract Type - will allow the generation of the appropriate integration file, which will contain all the necessary information for the selected DWH solution to use (read) the extracted data. The DWH Extract Type can be found in the [Extract Type](#) frame of the [Send To Server](#) window.

## Delete a Query

There are two ways to delete a query. The first is to open the query onto the NatQuery desktop, then select the Delete option from the File menu. The second is to select the query from the Open Query window, and click the Delete button in the lower left-hand corner of the Open Query window. In both cases, NatQuery will prompt the user to make sure that they really wish to delete the selected query.

Deleting a query has no effect on DDMs, extracted data, or any other part of the Environment Configuration – only the query specification itself is deleted.

## Delimiter Option

Delimiter Option – is found in the [Extract Type](#) frame of the [Send To Server](#) window. If either of the Download to Access or Download to Excel extract types are selected a delimiter option becomes available. The default delimiter for either the Excel or the Access extract types is a semi-colon (;), but the user may change this to whatever character is desired or needed. The user is strongly advised that care should be taken with this option, since specifying a delimiter that occurs in the extracted data could cause problems when using that data with other software (Access and / or Excel).

## Descriptor

Descriptor - is a term applied to a field that is usable as a “key” or “index” into a data source file, and represents a data access path into a given file. A descriptor field for a file is designated by a Database Administrator, and allows a file to be accessed in a very efficient and performance oriented manner.

## Dynamic Variables

[Dynamic Variables](#) - are special variables that allow their value to be changed dynamically each time the Query is generated. The two methods by which a Dynamic variable changes, are through an End-user prompted input or by being reset automatically such as a dynamic date substitution.

As an example of an End-user prompt dynamic variable, let us say that we need a Query that extracts data from a file that contains the field CITY and we only want to extract records that are associated with the CITY of "BOSTON". As described above, we might use a constant for this. This approach would work fine until we want to run the exact same Query again, but with a different value for CITY. While modifying or creating Queries is a simple process, we could equate the value of CITY to a dynamic variable (instead of a constant variable). Now when the query is generated, NatQuery will prompt the End-User to enter whatever value of CITY was required for the extract.

NatQuery also allows the use of dynamic variables that are set automatically. These automatically set variables relate to the current date, or a common variation of the current date. When a dynamic date variable is used, NatQuery sets the dynamic variable to the current date, using whatever date format was specified when the dynamic variable was created.

# NatQuery Users Guide

## **E mail option**

E-mail option – is a download option the user may select during the [Output Handling](#) process. For more information regarding Output Handling, please refer to its description, which is found within the description of the seventh icon, “[Send To Server](#)”.

## **Environment Configuration**

Environment Configuration - is a collection of files that when combined will allow NatQuery to intelligently generate Natural data extract programs from user-entered Query specifications. An Administrator builds an Environment Configuration by using an Administrative Version of NatQuery. Once built, an Environment Configuration can then be provided to other NatQuery installations to allow these other installations to generate extract programs as well. Please consult your administrator for more information with regard to this process.

## **Excel**

Excel™ is a “spread-sheet” tool developed by Microsoft – see [extract types](#)

## **Expression Variables**

**Expression Variables** - allows your Query to take fields (or other variables) and perform simple mathematical computations with them, thereby creating a result. As an example, let us say you have a field in a file that is named SALARY. For the purpose of your extraction, you would like to have an additional field that shows what this SALARY field would be if it were multiplied by .05. This is an example of where an expression variable would be used.

Please note that expression variables can have a format of Numeric, Packed Numeric, Date, or Time. An expression variable cannot have an alphanumeric format.

## **Extract Type**

Extract Type – offers the user the choice of program that will be used to download the desired data into a target such as Microsoft’s Excel or Access programs. The [Extract Type](#) frame is found on the “Send To Server” window that the user will encounter as they prepare to retrieve the required data. For more information, please refer to the description of the seventh icon “[Send To Server](#)”.

## **Favorite**

Favorite – the user has in the “[Query Description](#)” window an option to check a box marked Favorite. The user would check this box if they wanted to remind themselves this was a favorite query or that it was used often for some reason. For more information on this function, please see the section on the number two icon, “[Open Query](#)”.

## **Fields for Retrieval**

[Select Fields for Retrieval](#) is the window that will be populated with the fields the user has chosen from the Possible Fields to Select window of the Select Fields from FileName. These fields will be included in the data retrieved from ADABAS and the associated Field Names will serve as column headers when the download target is Excel or a similar business tool.

See also; [Include with Extracted Fields](#).

## **File Variable**

File Variables are variables that are defined by an Administrator against a specific file (DDM) such that these variables then become “attached” to the specified file. Also known as, "Local Variables".

See also [Global variable](#).

## **Filter on Flatten**

Filter on Flatten - NatQuery is sensitive to the situation where selection logic exists against a [recurring](#) field being [flattened](#). If this situation is detected, the user will now be prompted as to whether the records being flattened should be examined against the entered selection. If the user indicates that the Selection Logic should be applied, then only those flattened occurrences that adhere to the Selection Logic will be written out. If the user indicates that Selection Logic should not be applied, then all flattened occurrences will be written out.

# NatQuery Users Guide

## **File/Field Value**

The File/Field Value function allows an End-user to specify a Selection Logic statement that has, as its last operand, a reference to a file field, instead of a reference to a variable or a constant. For more information on the use of File/Field Values, see the description concerning [Selection Logic](#).

## **Flatten**

**Flatten** - when a [recurring](#) field is referenced in a NatQuery extract specification, the user is given a choice as to how NatQuery should handle the individual elements of a recurring field: A recurring field can be Concatenated or it can be Flattened. Flattening instructs NatQuery to handle each element of a recurring field in its own individual extract record (along with all other non-recurring fields being repeated), as well as assigning a unique index field for each element. Flattening of a recurring field means that for a single record being read from a data source file, multiple records will be written to the physical extract file.

See also [Concatenate](#), [Recurring Fields](#)

## **Format**

**Format** – is a term for the description of what data looks like. An example would be; two-column format, this describes data shown in two columns, or a numeric format describes data containing only numbers. The Modify or Define Query Variable windows allow the user to choose a variable format type: Alpha Numeric, Binary, Date, Integer, Numeric, Packed Numeric, and Time.

## **Help**

The Help section of NatQuery will attempt to provide guidance and understanding to the user through- out the query building process. Much of this document comes directly from the pages of the Help contents found in NatQuery. If the user has more in-depth questions with regard to NatQuery, its functionality, and capability, they should consult their ADABAS administrator. Help, can be access from the Tool bar at the top of the open NatQuery desktop, Help/Contents.

## **Global**

Pre-Defined Variables are of two types, Global and [File](#). Global Variables are variables that can be defined once by an Administrator for a given Environment Configuration, and once defined these Global variables can be either directly used by an End-User in the creation of Query specifications, or these variables can be used by an Administrator in the definition of File Variables.

## **Initialization Condition for VariableName**

The Define Initialization Condition for Variablename window is used to define the condition under which the variable Variablename will be initialized. If the variable is numeric, the default value will be zero. If the variable is alphanumeric, the default will be an empty string. Only when the specified condition is met will the variable set to the value specified in the appropriate window.

## **Intersect**

**Intersect** – is a data retrieval option available to the user and it is found in the Data Retrieval Options frame of the [Send To Server](#) window.

In the example case of EMPLOYEES and VEHICLES, the two files are linked together by the field PERSONNEL-ID. There will always be an employee record to match a record from VEHICLES, but the opposite is not true. There may be no record in VEHICLES that matches a given employee record. EMPLOYEES may have no car, one car, or more than one car. When a query is built that uses both of these files, NatQuery has no way of knowing whether to extract all of the EMPLOYEES records, or just the ones that can be linked to a vehicle (or vehicles). Specifying the All Records option will cause NatQuery to extract all EMPLOYEES records, regardless of whether or not there is a VEHICLES record (or records) associated with them. Specifying the Intersection Records option will cause NatQuery to extract only those EMPLOYEES records that have at least one VEHICLES record associated with them. The default is All Records.

# NatQuery Users Guide

There is a case where you will not be allowed to specify which option to use. That case is where there is Selection Logic against more than one file in the query. In this case, Intersection Records will be used by default.

## JCL / Script

JCL / Script – JCL means Job Control Language this specific language is used by NatQuery to automatically perform extractions against ADABAS and Natural when they are located on the mainframe. If ADABAS and Natural are on platforms such as Sun Solaris (UNIX) then NatQuery will require scripts for the automated extraction from ADABAS. JCL / Scripts are under the control of the administrator but reference to them may be encountered in some NatQuery error windows or special processing operations see “[e-mail](#) my data”.

## List Variable

[List Variable](#) - are special variables that are exclusive to usage with Selection logic, with this variable type allowing a single variable to be equated to multiple values.

## Lookup

Lookup - is a special type of file [Cardinality](#) within NatQuery that allows for one file (file #1) to be pointed at another file (file #2) for the purpose of having the fields from file #2 appear as though they are part of file #1 to an end-user. When examining the fields of file #1 for possible extraction, Lookup fields will appear as being part of file #1, and can be easily identified by a “> “ which precedes them in the field listing for file #1.

Lookups are defined by the NatQuery Administrator through the Administer File Relationships function, and are always considered an x: 1 relationship. See also [Cardinality](#), [standard](#), and [primary files](#) in this glossary.

## Multi-Value Field (MU)

Multi-Value Field (MU) - a type of recurring field that can occur within a file structure and that allows for a single field to occur up to 191 times for a single record. From an end-user perspective, a MU can be thought of as a list or an array of values. An example of a MU in a file structure would be a file structure that needs to store a street address. For most street addresses, a single field might suffice; however there are many instances when a street address needs more than just one field – it may need two or more street address fields to accurately store a specific street address. A Database Administrator (DBA) then, has a choice of how to store this information. The address lines could be stored in separate fields on a single file (street-address-1, street-address-2...). Or, they could also be stored as separate records into a separate file. Alternatively, the street address could be stored as a single field on a file with a MU that is able to occur several times.

For more information, see [Recurring Field Handling](#).

## NatCDC

NatCDC is NatWorks Change Data Capture tool that will assist in the use and management of your ADABAS Protection Log (PLOG). NatCDC was specifically developed for PLOG processing and the populating of Datawarehouse or DataMarts with the “Delta” (Changed) images of an ADABAS file. For more information on PLOG processing, please visit our web site at [http://www.natworks-inc.com/Products/Slide\\_Show/NatCDC/natcdc.html](http://www.natworks-inc.com/Products/Slide_Show/NatCDC/natcdc.html)

## Occurs

Occurs – this function can be found in the [Select Fields From FileName](#) window and again in the [Query Selected Fields](#) window. Occurrence information pertains to recurring fields ([MU's](#), [PE's](#) and MU's in PE's) also see [Recurring / Repeating Fields](#). Occurrence Information for MU's and PE's and MU's in PE's is an administratively installed parameter.

## Offset Report

The offset report generated by NatQuery is laid out as a table and gives starting and ending offsets (positions) of fields within the file. This is useful for users who need fixed-width fields in data files, and need to know the offset of those fields. An example of an “offset report” generated by NatQuery can be found below; the column

# NatQuery Users Guide

headers “Start” and “End” denote the position of the field in the row. “Length”, is the number of characters inclusive of Start and End. “Format”, tells the user what type of data they are receiving, in this example all the data is in alphanumeric format.

```
*
* NatQuery Offset Report
*
* Date: 3/22/2004   Time: 3:18:02 PM
*
* Version: 3.3.3
*
* FILE01: EMPLOYEES
*
```

ID	File	Field Name	Format	Length	Start	End
1	FILE01	FIRST_NAME	A	20	1	20
2	FILE01	NAME	A	20	21	40
3	FILE01	ZIP	A	10	41	50

## “Or” Logic

“Or” logic allows the user multiple specifications of unique values that will meet the selection logic criteria. As in: Employees with the last name of Jones living in Boston, New York, or Dallas.

## Periodic Group (PE)

Periodic Group (PE) - is a type of recurring field that contains a grouping of fields that can occur within a file structure and that allows for a group of fields to occur up to 191 times for a single record. From a user perspective, a Periodic Group Field can be thought of as a table. An example of a Periodic Group in a file structure would be a file structure that needs to store information about a person’s dependents. A person may obviously have no dependents or may have several dependents, with each of these dependents having a first name and birth date. A Database Administrator (DBA) then has a choice of how to store this information. The dependent information could be stored into separate fields on a single file (Dependent-1-first-name, Dependent-1-birth-date, Dependent-2-first-name...). This information could also be stored as separate records into a separate file. Alternatively, the dependent information could be defined as a PE GROUP within a single file, with group containing the fields Dependent-first-name and Dependent-birth-date, with these two fields being able to occur several times.

For more information, see [Recurring Field Handling](#).

## Phonetic Descriptor

Phonetic Descriptor - is a special type of descriptor that utilizes a specialized naming algorithm that allow for searches to be performed against an ADABAS files on text strings that are phonetically similar.

## Platform

NatQuery has been developed for use on the Personal Computer (PC). Because ADABAS and Natural can be found on several platforms, including OS/390, VMS, UNIX, and other “Open Systems”, the various NatQuery windows the user will encounter may differ from one operating system to the next.

## Pre Defined Variables

NatQuery allows an Administrator to pre-define one or more [variables](#) such that these variables become associated with a specific file / DDM. Through this definition process, a user of NatQuery is then allowed to immediately reference / select these pre-defined variables at the time they select fields from the associated file,

# NatQuery Users Guide

without the need of the user actually defining these variables. A predefined variable might be used when one or more fields in a given file require some sort of manipulation in order to become useful to the user.

For example, assume that there is a file / DDM with one or more numeric fields that are typically used in a calculation to produce a value that is then immediately meaningful to a user. While the user is certainly able to define an [expression variable](#) to produce the calculated value when the user builds a query extract – the user would be required to redundantly define this expression variable every time a new query was built that used that file and that had a need of the calculated value. This amounts to redundant work on the part of a user.

To address this redundancy, the Administrator now has the option of defining one or more variables that then become associated with a given file, such that when a user references this file for designating fields for extraction, these variables become selectable in the same manner that fields from the file are selectable. When the user selects one of these pre-defined variables, the underlying definition of these variables is automatically included into the Variable definitions of the query being handled, completely removing the need of the user to have to define these variables.

See also; [Global](#), and [File](#) (Local) variables

## [Predict Information](#)

Predict information is available from three windows. By highlighting a given field in the [Query Selected Fields](#) window, and clicking the Predict Info button, Predict Information will become available. By right clicking over a selected field in either the [Select Fields from Filename](#) window, or the [Selection Logic](#) window, a user will also be able to review the Predict information. The purpose of the resulting Predict Information window is to display descriptive information about a specific field.

Using the Predict function, a user can also find information regarding [Occurrences](#) within a field. A description of each MU, PE, and all MU's in PE's will be explained within the Predict generated report.

End-users are not allowed to change information in this window. Administrators may change the information that is displayed for a given field simply by changing the text in this window and clicking on the OK button, which saves it to be recalled later by either administrators or end-users.

Predict may not be available for your environment.

## [Primary Files](#)

[Primary Files](#) – The files that are made available to the End-User in the “[Select File](#)” window. The availability of these “Primary Files” to the user is made by an administratively controlled operation during the “Environment Configuration”. The user will be allowed to select up to five of these primary files for their use when building any given query. After the first file is selected, the files that are available for selection will be limited to just those files that have a valid relationship to the currently selected files.

## [Production Request Processing](#)

The Production Request Process - template is the JCL template that will physically execute a user-specified data extraction request. The Production Request Process template being built is generic in its definition, and is made specific to a given user and a given user's User Request File using dynamic substitution parameters. These parameters are designated by an “&&”. The building of JCL is an Administrative function, for more information on JCL and the templates that NatQuery uses please see your Administrator.

## [Program Name](#)

Program Name – the default name automatically produced by NatQuery for a new query. At the End-user's discretion, the system provided Program Name could be over-riden by entering a Program Name of the End-

# NatQuery Users Guide

user's choice. NatQuery utilizes the Program Name field as the PC filename of the query specification (with an extension of .QRY), as well as being used for End-user identification of any given query. The Program Name is found (or modified) in the "[Query Description](#)" window and can contain a maximum of eight characters. It is mandatory that a Program Name be in place before query processing can take place.

## **Recurring / Repeating Fields**

Recurring / Repeating Fields - are fields that have the capability of "occurring" multiple times for a single record. There are two types of recurring fields. Multi-Value Fields, or MUs, are fields that can occur up to 191 times for a record, and this structure is best thought of as a list or an array. Periodic Group fields, or PEs, are a group of fields that as a group can occur up to 191 times for a record, with this structure best thought of as being a table. Additionally, an MU can be placed inside of a Periodic Group, allowing for a multi-dimensional table.

See also [Multi-Value Field \(MU\)](#), [Periodic Group \(PE\)](#) in this glossary

## **Redefinition Variables**

[Redefinition Variables](#) - allows you to "cut-up" a field from a file into a smaller "piece". As an example, let us say we have a field called LAST-NAME on a file, and this field is 20 characters long. Of course, you can just select this field for your extract and have the whole field become part of your data extract. For the sake of argument however, let us say that you only need the first 5 characters of the field LAST-NAME returned with your extract data. In this case, you can create a redefinition variable that represents just the first 5 characters of the field LAST-NAME, with this redefinition field then being available to be part of the data extract.

Please note that redefinition variables can only occur against fields that have an alphanumeric format. For this current version of NatQuery, any field that does not have alphanumeric format cannot be redefined with a redefinition variable.

## **Re-sequencing**

Re-sequencing the order of fields as they will be seen in the extract target can be accomplished using the Up and Down buttons located on the right hand side of the [Query Selected Fields](#) window.

## **Retrieval Limits**

Retrieval Limits – the user may request a limited retrieval by choosing Limited, or they may opt for No Limit in the retrieval of their data request. This option is found in the Retrieval Limit Options frame of the [Send To Server](#) window

## **Search**

## **Standard File Relationship**

A "standard" File Relationship defines the usual relationships that exist between two "primary" files. Using a "standard", File Relationship means that End-User will select fields from these files by handling each file independently. For more information on File relationships, please see [cardinality](#), [lookup](#), and [primary files](#).

## **Summary Processing**

The Summary Processing function allows for the specification of how "Detail" data that will be provided by a given extract may be subsequently processed into "Summary" data.

## **Super-Descriptor**

Super-Descriptor - is a term applied to a collection of fields that when combined together become usable as a "key" or "index" into a data source file, and represents a data access path into a given file. A Super-Descriptor is typically comprised of multiple individual fields, and is usually designated by a Database Administrator. A Super-Descriptor allows a file to be accessed in a very efficient and performance-oriented manner.

## **Table**

A Table is an orderly set of values given in a multi-dimensional format; a simple example would be an Excel spreadsheet with both rows and columns. A working example can be found under the description; [Normalize my Data](#). For more information, please refer to [Specifying Table Occurrences](#).

# NatQuery Users Guide

## Target

Target or [Extract Type](#) – is a term for the name of the place where the user will download their data. NatQuery can download data directly into many useful places such as Microsoft’s office tools Excel and Access. Other “targets” include the users e-mail client, a PC file, XML, or other databases such as DB2. The list of targets a user will encounter will be dependent on the server platform where ADABAS is installed.

- Data Download to PC File
- Data Download into Excel
- DWH Software Extract
- SourcePoint Extract program
- Data Download into Access
- RDBMS Loading
- Data Download into XML
- Any2XML Integration (Extract data to the web)
- Extract to XML on the Server

## Variable

[Variable](#) - is a Query element that allows for the fields which can be extracted by NatQuery to be extended beyond just the fields that are available from a given data source. A variable can be thought of as a field or “container”, with the contents of a container being defined by a user to be one of six types: [Redefinition](#), [Expression](#), [Constant](#), [List](#), [Dynamic](#), and [Compress](#). With these six types of variables, data information can be both supplied to a Query, as well as being derived by the Query.