

4D Ajax Framework

*Admin Reference Guide
Windows/Mac OS*



CHAPTER 1: OVERVIEW	4
INTRODUCTION	4
AT FIRST SIGHT	5
THE CLIENT ENVIRONMENT	7
CONTROL PANEL.....	8
CHAPTER 2: CREATING QUERIES VIA THE QUERY MANAGER	17
INTRODUCTION	17
GETTING STARTED	18
CREATE A CUSTOM QUERY	19
SET DYNAMIC QUERIES	22
USE 4D'S BUILT-IN QUERY TEMPLATES	23
FINISHING UP	24
CHAPTER 3: USING THE ACCESS CONTROL TAB.....	26
INTRODUCTION	26
DIFFICULTY METER.....	27
GETTING STARTED: THE BASICS	28
ADVANCED ADMINISTRATING	34
CHAPTER 4: DEVELOPER DEFINED WINDOWS AND YOU.....	38
INTRODUCTION	38
5 STEPS TO CREATING A DDW	39
DIFFICULTY METER.....	40
STATIC LINK DDW	41
DYNAMIC LINK DDW	42
HTML BLOB DDW	45
TEXT MESSAGE DDW	47
ASSOCIATING DDWS.....	49
CHAPTER 5: DASHBOARDS.....	53
INTRODUCTION	53
DASHBOARD FEATURES.....	54
THE DASHBOARD EDITOR	61
CHAPTER 6: DATA GRID.....	65
GENERAL INTERFACE	66
ADD/DELETE RECORDS	66
PRESET QUERIES	66
INLINE EDITING	68
SEARCH.....	69
STATUS BAR.....	72
PICTURE PREVIEW	72
CHAPTER 7: OFFLINE MODE	73
REQUIREMENTS.....	73
OFFLINE MODE IS NOT A FULL CLIENT	73
USING OFFLINE MODE	75
ADMINISTRATOR: ENABLE TABLES FOR OFFLINE MODE	75
END USER: GOING OFFLINE AND BACK.....	75
END USER: OFFLINE ON THE IPHONE.....	76
CHAPTER 8: IPHONE DATA ENTRY.....	78

ENABLE A TABLE FOR IPHONE..... 78
THE OUTPUT FORM 78
CHOICE LISTS 79
OFFLINE MODE ON THE IPHONE..... 81

Chapter 1: Overview

Introduction

The 4D Ajax Framework provides the most robust and convenient means of enabling web client access to 4D databases. With the evolution of lightweight, agile, and dynamic web-based applications in Web 2.0, 4D developers now have the ability to harness that power without having learned the ins and outs of JavaScript, CSS, and other web technologies. For existing 4D developers, the transition to supporting web-based clients can be quick, seamless, and exciting thanks to the tools, libraries, and built-in services provided by the 4D Ajax Framework. Below is just a sampling of reasons why the 4D Ajax Framework is the revolutionary product we believe it to be:

Live Data: With Ajax on your side, the user experience for your clients is live and dynamic. Updating data will be instantly refreshed in the client environment without having to refresh the full page in the web browser.

Easy Integration: Apprehensive about integrating the 4D Ajax Framework to your complex 4D structure? The 4D framework makes this a calming process requiring you to not write code and to only install a component.

Live Scrolling: Part of the live data experience, information gets pulled from the server as you scroll through a table. No need to wait for preloading records from the server. Data is pulled to the client on demand eliminating the hassle of stale and static pages of data.

Searching: Find anything from the database as quickly as you type. Results start displaying as soon as the user types the first letter.

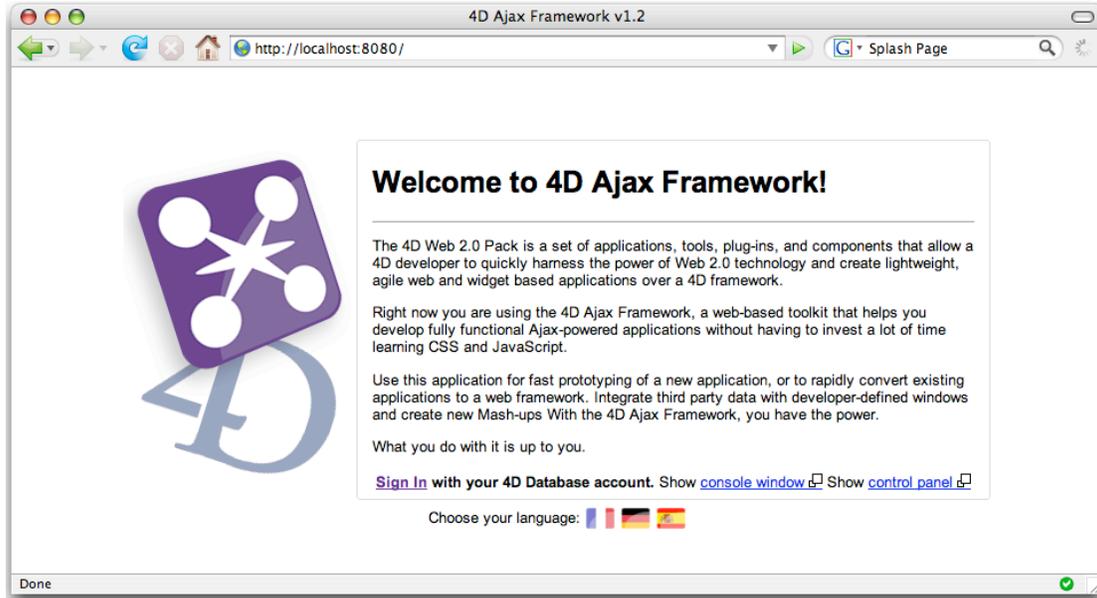
Tabs: Clients can save searches on the selection tables as tabs, which can be referenced or deleted at any time.

Built-in Data Services: The Data Services layer of the 4D Ajax Framework acts as a communication layer between the 4D web server and the web client. This Data Services layer provides many built-in features such as live data streaming, the handling of record locking, query handling, data organizing, etc.

These features and more make the 4D Ajax Framework the perfect solution for 4D developers who want to create a dynamic web-based application and a rich user experience in a 4D environment.

At First Sight

After a successful installation of the 4D Ajax Framework component and associated files, starting your web browser will launch the following splash page:



Viewing this page tells us that so far that setup has been a success. You have successfully installed the 4D Ajax Framework component to your database, and the proper web folder has been copied to the correct locations relative to your database structure. Congratulations, you are well on your way to adding the power of the 4D Ajax Framework to your database.

On the splash page we have links for Sign In and for the Console Window.



Sign In

This is where clients identify themselves as they log into your web server. The accounts, passwords, and privileges are set up by you in the

development environment so only users you designate are provided access. 4D's built-in password system can be implemented here or if you have your own password system that is supported as well.

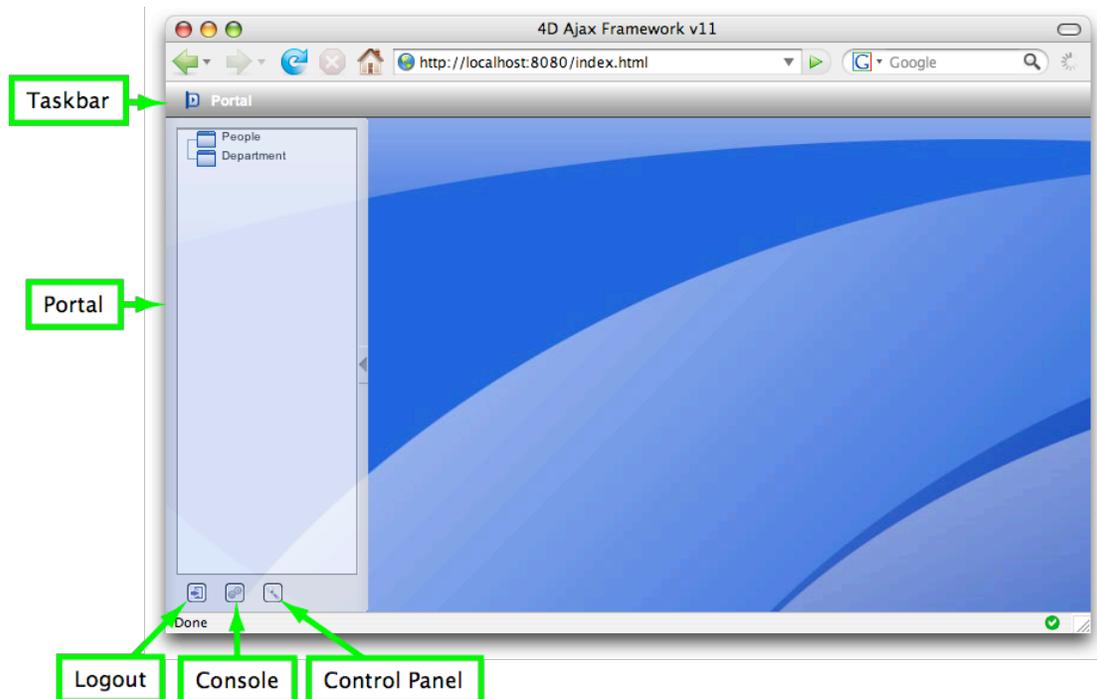
Console Window

When clients connect to a web server over the World Wide Web, data is sent to the server that identifies the user. This data is known as the User Agent, which tells the server such information as the client's web application, web application version, host operating system, and language, among other things. As a 4D web server developer, this type of information can come in handy when debugging unwanted behavior in your database or application.

Not only does the Console Window display User Agent information, it also displays a log of communication between the web server and the client. With the Console Window left open, if you then sign in or perform other actions in the client environment you will be able to see a log of calls between the server and client that, once again, can come in handy in debugging.

The Client Environment

The 4D Ajax Framework lets you choose from different layers of customization. For developers who like to plug-in right out of the box they can have their clients use the built-in front-end to act as their application client. This option provides the least amount of customization in the least amount of time when creating a web client for your 4D database. After a successful login at the Sign In window, below is a snapshot of the default environment that the clients are presented with.



Portal: The Portal toggles the ability to show/hide a list of selections. When we say selections we do not just mean Tables. We also include Views and Developer Created Selections (DCS). As administrator of this environment, you can assign what selections each client is able to access. By default the Portal lists the first 10 selections of the database.

Logout: This is where clients log out of their session with the web application.

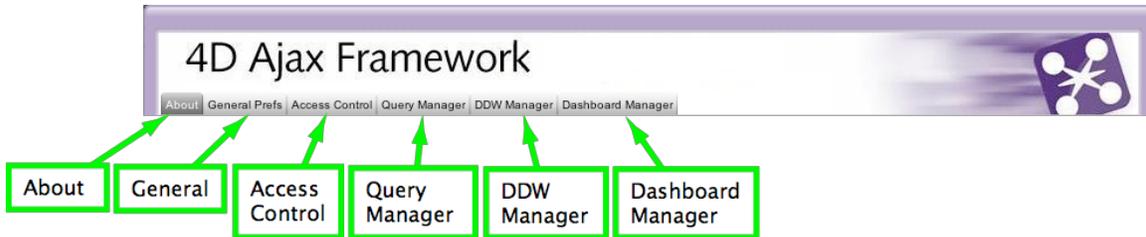
Console: The Console behaves in the same way it does at the Splash screen. The only difference is that this button provides the ability of the Console to be loaded up in the client environment.

Control Panel: This button is for administrators giving them the ability to, among other things, change the client environment style, specify client access, etc. Only users with administrative privileges are

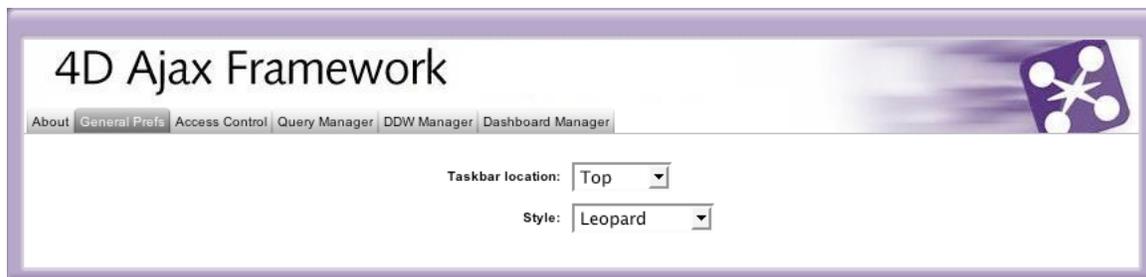
allowed to access this feature in the client environment. A more detailed overview about the administrative settings follows in the next section: Control Panel.

Control Panel

Clicking the Control Panel button in the client environment gets us to the following window with 4 tabs: General Prefs, Access Control, DDW Manager, and Query Manager.



General Prefs



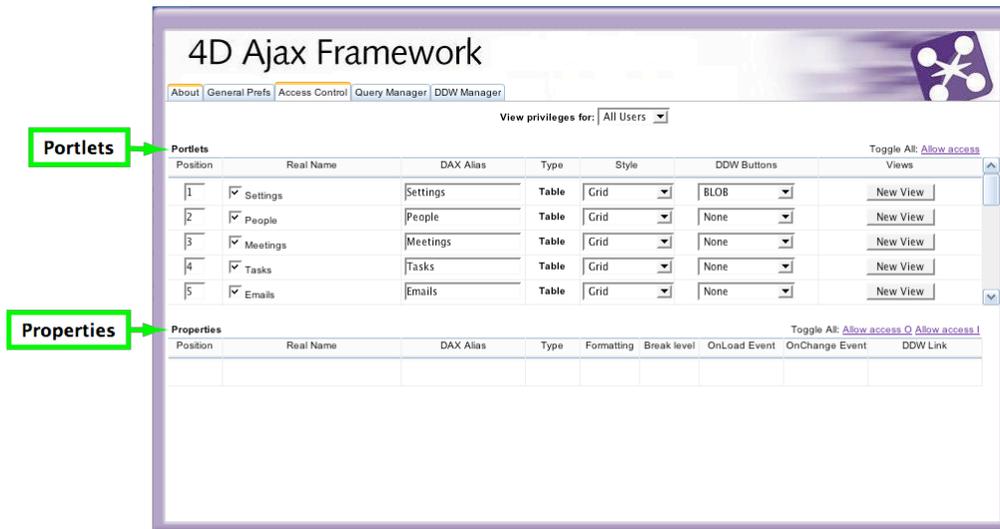
This tab allows for high-level "look and feel" changes to the client environment.

- *Taskbar Location:* Set the taskbar to the top or bottom of the client environment area in the web browser.
- *Style:* The 4D Ajax Framework comes with predefined themes for the client environment. Three (3) themes are loaded by default and more can be optionally loaded. Many of the themes are based on the look of some notable operating systems in use today. Feel free to choose whichever theme fits your style. Or, you can also create your own themes. All functionality is the same regardless of what style is chosen.

Access Control

The Access Control tab is where to go to define privileges for each user. Here you can set which selections they can access, what fields they can see, what

style the tables are presented as, etc. As an administrator this is where most of your administrative duties are performed.



Before we delve into the access settings that are found further down this table, take a look at the **View Privileges for** pull down menu at the top.

View privileges for: All Users ▼

If you have users and groups already set up in your database, this is where you would go to select users and define access privileges.

Portlets: This section of the Settings tab lists all tables that are created in your structure. It can also list other objects such as Views, Developer Created Selections (DCS), and Developer Defined Windows (DDW).

You can select/deselect all Selections that can be viewed by this particular user by clicking the **Toggle All: Allow access** link.

Position	Real Name	DAX Alias	Type	Style	DDW Buttons	Views
1	<input checked="" type="checkbox"/> Settings	Settings	Table	Grid	None	New View
2	<input checked="" type="checkbox"/> People	People	Table	Grid	None	New View
3	<input checked="" type="checkbox"/> Meetings	Meetings	Table	Grid	None	New View
4	<input checked="" type="checkbox"/> Tasks	Tasks	Table	Grid	None	New View
5	<input checked="" type="checkbox"/> Emails	Emails	Table	Grid	None	New View

- **Position:** This field defines where each selection is listed in the Portal (found in the client environment). A selection set to position '1' is the first selection listed at the top of the Portal list.
- **Real Name:** For tables, this is the actual name of the table in your structure. This information can come in handy if there is any confusion as to what table you are referencing. Views and DDWs can be listed here as well, although no corresponding object is created in the structure. This column also has a checkbox for each table. A box that is checked means that the table will be available to this user in the Portal. Conversely, an unchecked box means that the table will not be listed in the Portal list.
- **DAX Alias:** This allows you to show user-friendly names in the Portal, instead of the Real Names.
- **Type:** Objects in the Portlets list can be of type Table, View, DDW, or DCS. If the 4D Ajax Framework was recently added to your 4D database then every object in this list should initially be of type Table. After some customization, Views, DDWs, and DCSs can be created for the client environment.
- **Style:** Selections in the Portal can be displayed in different ways, depending on what style you feel best suits the information. The options are Grid, Calendar, Data Tree, DataMatrix, and ImageMatrix. For example, if you have date fields in your selection you may find the Calendar display style beneficial for your clients. Examples follow:

Grid

First	Last	State
Alfred	Bates	New York
Peter	Parkers	Washington
Steve	Jhobs	Ca

Data Tree

Task
Set up conference call
Bring Wii Controllers

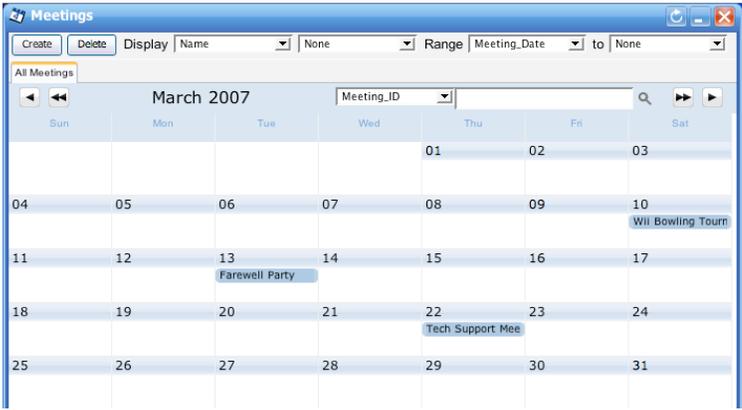
Image Matrix



DataMatrix



Calendar



- *DDW Buttons:* After Developer Defined Windows (DDWs) are created this is where the administrator can assign the DDW to a selection window. When the selection window is loaded in the client environment, your DDW is presented as a button. This gives clients the ability to select records in the selection window and then perform your method by clicking the DDW button.
- *Views:* This is where Views are created. A View acts much like a virtual table. When we have related tables, a View has the ability of becoming a new virtual table able to display fields from any table related to the table currently selected. This allows you to present data in a way that makes sense to users, instead of just what's in the database.

Properties: For every Table there can be one or many fields. For each Table selected in the Portlet above corresponding fields are listed here in the Properties section. This same concept also applies to fields of Views, Developer Defined Windows (DDW), and Developer Created Selections (DCS) in the Portlet.

Position	(O S)	Real Name	DAX Alias	Type	Formatting	Break level	OnLoad Event	OnChange Event	DDW Link
1	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	People First	First	A	None	None	<input type="checkbox"/>	<input type="checkbox"/>	None
2	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	People Last	Last	A	None	None	<input type="checkbox"/>	<input type="checkbox"/>	None
3	<input type="checkbox"/> <input checked="" type="checkbox"/>	People Age	Age	0%	None	None	<input type="checkbox"/>	<input type="checkbox"/>	None
4	<input checked="" type="checkbox"/> <input type="checkbox"/>	People City	City	A	None	None	<input type="checkbox"/>	<input type="checkbox"/>	None
5	<input checked="" type="checkbox"/> <input checked="" type="checkbox"/>	People State	State	A	None	None	<input type="checkbox"/>	<input type="checkbox"/>	None

- *Position:* As clients choose selections in the client environment, this setting defines where each field is listed in the selection window. A field set to position '1' is the first field listed to the left of the selection window.
- *Access:* Introduced in version 1.2, the *O* and *I* checkboxes determine whether or not the field is displayed in the Output and Input form respectively. This is a useful feature should you want to hide certain fields from certain user groups.
- *Searchable:* Introduced in 4D Ajax Framework v11 Release 1 (11.1), this check box makes fields searchable. If a field is searchable, then the search toolbar for the Data Grid appears. This toolbar allows end-users to build their own queries based on the searchable fields.
- *Real Name:* For fields, this is the actual name of the field as stated in the table of your structure. This information can come in handy if there would be any confusion as to what fields you are referencing. This column also has a checkbox for each field. A box that is checked means that the field will be available to this client in the Portal. Conversely, an unchecked box means that the field will not be listed in the Portal list.
- *DAX Alias:* This is the alias of the corresponding field that is shown to the user in the selection window. The administrator is given the option of presenting a field with a unique name depending on which client is logged in.

- *Type*: This displays the type of data that is represented in this field. Fields can be of type alphanumeric text fields, pictures, Booleans, BLOBs, etc.
- *Formatting*: The 4D environment allows field information to be formatted in certain ways depending on the data type of the field. For example, text fields can be formatted to upper or lower case form, or date fields can be formatted to MM-DD-YYYY or DD-MM-YYYY form.
- *Break Level*: This option proves useful in situations dealing with Data Trees. Some data fields may be repeated on multiple lines when each field refers to different pieces of information. For example, suppose you have three tasks assigned to one meeting. In your selection window, you may see three lines of tasks assigned to the same meeting. This seems inefficient since the same meeting is repeatedly listed on those three lines as well. Break Levels organize this display by making the meeting field displayed on one line, and each task revealed under a drill down icon. In this scenario we would set our Break Level to 1 in the Meeting field.

Life Before Break Levels

Meeting	Task
Quarterly Staff Meeting	Print Status Report
Quarterly Staff Meeting	Buy Milk and Cookies
Quarterly Staff Meeting	Bonus Distribution Poster
Company Meeting	Bring Wii Controllers
Company Meeting	Set up conference call

Notice how the Meetings are repetitive.

Life After Break Levels

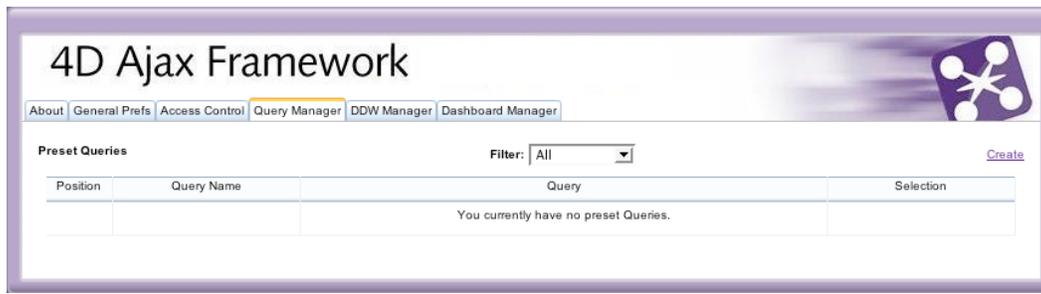
Meeting	Task
Company Meeting (2)	Set up conference call
	Bring Wii Controllers
Quarterly Staff Meeting (3)	Bonus Distribution Poster
	Buy Milk and Cookies
	Print Status Report

Information simplified.

- *OnLoad Event*: When users in the client environment create a new record or double-click a record for edit, the Input form appears which allows the record to be edited. This window appears at the same time the OnLoad Event occurs. For developers this may be an opportune time to, for example, automatically fill in date fields with the current date once the sheet window is loaded.

- *OnChange Event:* Here you can call a method every time data in a field is updated. This is the perfect place, for example, to call a validation method when clients are entering their emails in a database. When users enter data in the field your method would validate their entry as they typed along.
- *DDW Link:* Here Developer Defined Windows (DDW) can be linked to fields in your selections. When this occurs, the fields become hyper-linkable. Upon clicking the field link in the client environment, your DDW is loaded. For example, if you have an address field you can make it so that when the client clicks on the address link a new window is opened displaying that same address in a Google Maps HTML window.

Query Manager



The Query Manager enables a developer to save predefined queries for users in the client environment. This means that when selections are loaded from the Portal, their selection windows will open up with predefined queries saved as tabs. This allows you to increase convenience for users when working in specific Views.

When creating a query we are presented with the following in the Query Manager tab:

Create a preset query for this selection:

Create a new query:

Set dynamic queries:

Use this template: and apply it to this field:

- *Create a preset query for this selection:* This pull down menu lets you select the table to link with the query.

- *Create a new query:* This option is selected when you would like to create a customizable query. In the text field you give the query a name, which will be displayed for the client in the selection window.
- *Set dynamic queries:* This option saves queries based on distinct values from the field chosen in the pull down menu. For example, suppose you have a music library database and one of your fields in your tables is 'Music Genre'. By choosing the 'Music Genre' field in the pull down menu for this option, when the user loads this table they will see a tab loaded for 'Jazz' as well as other music genres values you have entered in this field.
- *Use this template:* The 4D Ajax Framework comes with a set of built-in templates that you can use when building queries. You can, for example, use the A-Z template for a 'First Name' field so that when the selection window is loaded records are organized by first name with each letter of the alphabet as a tab. You can also add your own templates as well.

DDW Manager



As a developer you are not restricted to the windows and display style of the windows pre-created in the client environment. Developer Defined Windows (DDWs) give the developer the power to call windows displaying information defined by your own code.

Title

Content

Type

Associated to

- *Title:* This is the name of the DDW that can be referenced in the DDW Buttons or DDW Link field in the Access Control tab, or in the Portal in the client environment.
- *Content:* A web URL or the name of one of your methods is entered here. When a DDW is loaded this is where the 4D Ajax

Framework references to determine the contents of the DDW.

- *Type:* This option determines how your DDW will be accessed. 'New Window - Link' enables your DDW to be loaded in a new window in the client environment. 'New Window - HTML Method' says that the 4D Ajax Framework will open a blank HTML window for you, but the code you call in the *Content* field will fill contents of that window. 'Status Message - Text Method' will load your DDW in a status message.
- *Associated to:* DDWs can be loaded like a selection in the Portal of the client environment. Or, they can be assigned to specific selections and fields by way of the DDW Buttons or the DDW Links in the Access Control Tab. Choose 'Portal' to associate your DDW to the Portal, or choose 'Other' to associate it to the Control Tab.

Chapter 2: Creating Queries via the Query Manager

Introduction

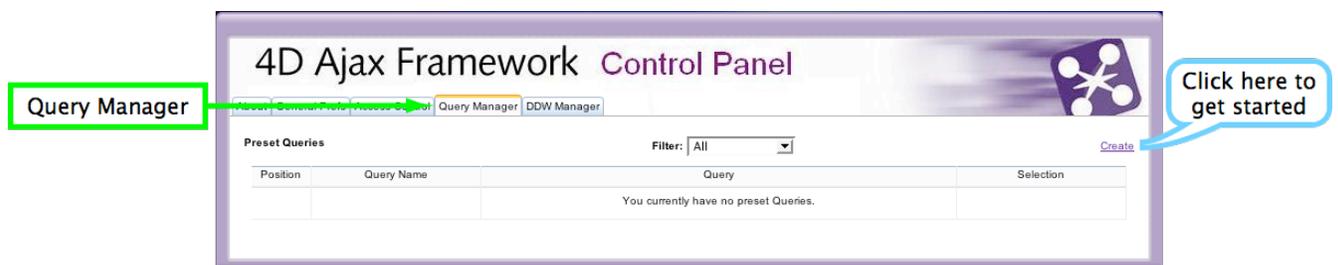
Welcome to the authoritative (and only existing) guide to creating queries in the Query Manager for the 4D Ajax Framework. This guide assumes that the reader has a familiarity with the 4D Ajax Framework client environment. Ready to get your feet wet? Good, let's begin.

Requirements

Before delving into to the Query Manager, let's make sure everything is set with the 4D Ajax Framework.

- Make sure the 4D Ajax Framework is properly installed to your database (see the "4D Ajax Framework Install" document).
- Log in as Administrator. (*Tip:* For administrative access the 4D Ajax Framework defaults with login "Administrator" with no password. If you are otherwise rolling out your own password system you must login with your own administrative account login and pass.)
- Click the Control Panel button.
- Click the Query Manager tab.

The following is a screen shot of what you should see.

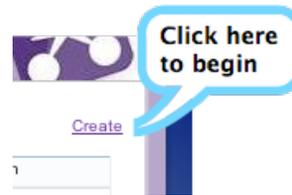


Congratulations! If you've gotten this far setup has been a success. Sure, an empty list of queries is not so exciting now but we promise that once you've harmlessly set up some queries you'll be able to proudly show your list to all your friends and coworkers.

Note: Sample screenshots are provided using the 4D Meetings Demo database. Feel free to launch it and follow along.

Getting Started

Click the Create link to create a query.



You will then be presented with the following:

A screenshot of a 'Preset Queries' dialog box. On the left, three options are listed in green-bordered boxes: 'Create a Custom Query', 'Set Dynamic Queries', and 'Use Built-in Queries'. Green arrows point from these boxes to the corresponding radio buttons in the form. The form includes a 'Create a preset query for this selection:' dropdown menu set to 'None', a 'Create a new query:' text input field, a 'Set dynamic queries:' dropdown menu set to 'Not Available', a 'Use this template:' dropdown menu set to '0-9', and an 'and apply it to this field:' dropdown menu set to 'Not Available'. A 'Next' button is at the bottom right. A 'Cancel' link is visible in the top right corner.

Egad! This may seem a bit intimidating at first but no need to be apprehensive. Initially the environment may seem a bit complicated, but I assure you things are not as daunting as they may seem. First, explanations are in order.

There are three ways to create a query:

- **Create a Custom Query:** Create queries based on your own requirements. ★★★
- **Set Dynamic Queries:** Create queries based on values already present in your database. ★
- **Use 4D's built-in query templates:** 4D provides built queries, such as *A-Z* or *US Regions*. ★

(Further information can be found in the *Query Manager* section of *Chapter 1: Overview* in this document).

At this page of the Query Manager, you are given the opportunity to decide what kind of query you want to make. Three options are presented for you and thus, what follows are three subsections in this chapter that explain each option.

Based on the difficulty meter (★★★★ = the most difficult), you may want to skip ahead and try options #2 & #3 before attempting method #1 (**Create a Custom Query**).

Create a Custom Query (★★★)

- Choose the selection you want the query to be performed on.

Create a preset query for this selection: None

Query Name:

Bound To: Not Available

- Make sure Create a new query is selected. Give the query a name. Click Next.

Create a new query:

- The following screen appears:

Preset Queries

Name: TPS Reports Bound To: Tasks

Sorting: Field: Task_ID Order: Asc

Maximum number of records:

Add a condition:

Link: and Field: Task_ID Operator: equal Value: is a method: Add

Conditions:

Link	Field	Operator	Value

Save

Take a moment to get familiar with this area if you have not already. The query name you just entered and the selection your query will run on are listed at the top. The rest of the window can be divided into 3 areas: 1) *Sorting Area*, 2) *Place to add conditions*, and 3) *List to display saved conditions*.

- Sorting Area:** A good place to start is in the sorting area. Choose the field you want the query to be sorted by and set it to Ascending or Descending order.

Sort by the field chosen here

Set it to Ascending or Descending order

Sorting: Field: Task_ID Order: Asc

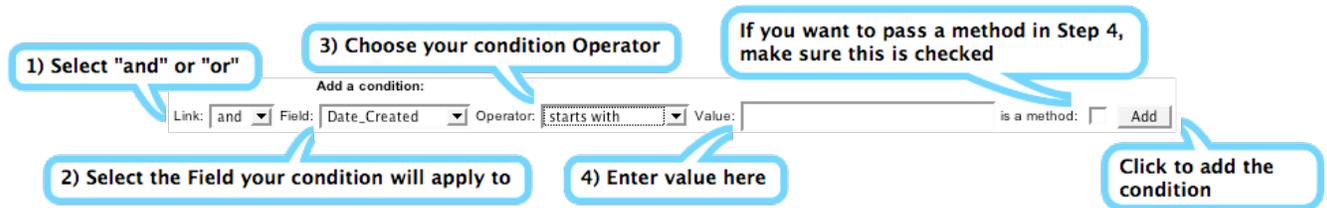
Maximum number of records:

Set the maximum number of records to be displayed

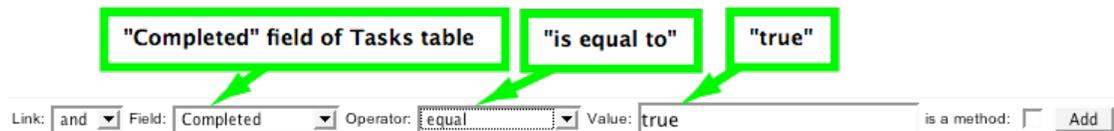
You can also set a limit as to how many records will be displayed. To set a limit check the *Maximum number of records* check box, and then enter the limit in the text field next to it.

Sorting is not necessary for you to create queries, thus these fields can also be left alone.

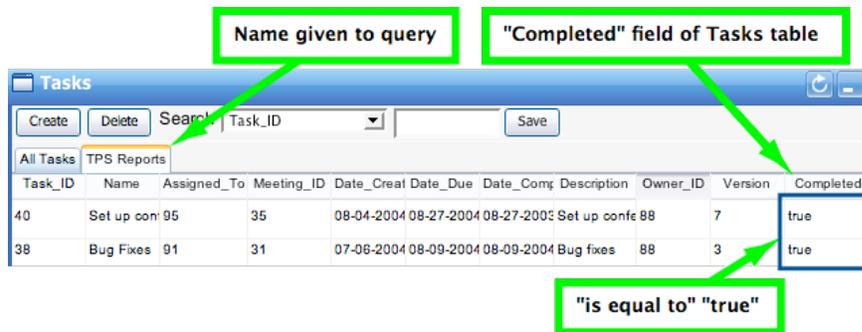
- **Add Conditions:** For experienced 4D developers this area may seem familiar. Like the Query Editor in the 4D user environment, conditions for your query can be added here.



Here is an example with the 4D Meetings demo database. We are using the Tasks table and are making a query for completed tasks.

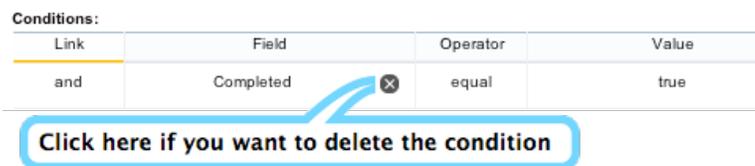


Here is the result when opening the Tasks table from the Portal.



List to Display Saved Conditions: When query conditions are added results are listed in this area.

Below is a screen shot of what this area looks like after adding the condition in the 4D Meetings example above.



Feel free to link more conditions to give your query added complexity.

Once your conditions are added, click the Save button to go back to the first page of the Query Manager tab.



You are encouraged to proceed with creating more queries. If you are done, please proceed to section ***Finishing Up***.

Set Dynamic Queries (★)

Dynamic queries are queries driven by data values in your fields. Let's use an example from the 4D Meetings demo database to see dynamic queries in action.

The screenshot shows the 'Create a preset query for this selection' configuration page. A green box highlights the 'Set dynamic queries' checkbox, with the text 'Make sure this is checked'. A blue callout '1) Choose the selection' points to the 'Tasks' dropdown in the 'Create a preset query for this selection:' field. Another blue callout '2) Choose the field' points to the 'Name' dropdown in the 'Set dynamic queries:' field. The 'Use this template:' field is set to '0-9' and 'and apply it to this field:' is set to 'Task_ID'.

Click the Next button to reload the Query Manager page. This is what you should see:

The screenshot shows the 'Query Manager' page with a 'New query added' callout pointing to a new entry in the 'Preset Queries' table. The table has columns for Position, Query Name, Query, and Selection. The new entry is at Position '-', Query Name 'Dynamic Queries', Query 'Data Driven Tabs based on Name', and Selection 'Tasks'. A 'Filter: All' dropdown and a 'Create' link are also visible.

Position	Query Name	Query	Selection
-	Dynamic Queries	Data Driven Tabs based on Name	Tasks

(For more information about the contents of this above area go to the **Finishing Up** section of this chapter).

Now load your selection from the Portal. In our 4D Meetings demo example, we would be opening the Tasks table. Below is screen shot of what you should see:

The screenshot shows the 'Tasks' table interface. A green callout box at the top says 'These queries were taken from values in the Name field of the Tasks table'. Below it, the 'Task_ID' dropdown menu is open, showing a list of task names: 'All Tasks', 'Bonus Distribution Poster', 'Bug Fixes', 'Set up conference call', and 'Tradeshaw Flyers'.

As you can see, dynamic queries are based on distinct values from any field of your choosing.

Use 4D's built-in query templates (★)

To understand 4D's built-in query templates perhaps it would be simplest to test them out first and see the results. The following is an example using the 4D Meetings demo database.

The screenshot shows the 'Create a preset query for this selection' dialog box. A green box highlights the 'People' selection dropdown with the text 'Make sure this is selected'. Three blue callout boxes provide instructions: '1) Choose the selection' points to the 'People' dropdown; '2) Choose the template' points to the 'A-Z' dropdown; and '3) Choose the field' points to the 'First_Name' dropdown. Other fields include 'Create a new query: Query 2', 'dynamic queries: Person_ID', and 'and apply it to this field: First_Name'.

After clicking the Next button, you are brought to this page in the Query Manager tab (for more information about the contents of this area go to the **Finishing Up** section of this chapter):

The screenshot shows the 'Query Manager' tab in the 4D software. A green box highlights the title 'List of queries created using 4D's A-Z template'. The 'Preset Queries' table is displayed with the following data:

Position	Query Name	Query	Selection
-	#	First_Name starts with 0 First_Name starts with 1 First_Name starts with 2 First_Name start	People
-	A	First_Name starts with A	People
-	B	First_Name starts with B	People
-	C	First_Name starts with C	People
-	D	First_Name starts with D	People
-	E	First_Name starts with E	People
-	F	First_Name starts with F	People

Now go to the Portal and open the selection your query is performed on. For our 4D Meetings example, open the People table.

The screenshot shows the 'People' table in the 4D Portal. A green box highlights the query tabs with the text 'These query tabs were created'. The tabs include 'All People', '#', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z'. The table data is as follows:

Person_ID	First_Name	Last_Name	Email_Addr	Phone	Fax	Machine_O	Login_Nam	Multiple_Ac	Add_To_Li
88	Administrat		admin@4d			wwdc	4dmeetings	true	true

Click on any tab to see records that match each query. For instance, clicking the 'B' tab in our example brings us Brandon's record.

All People	#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Person_ID	First_Name	Last_Name	Email_Addr	Phone	Fax	Machine_O	Login_Nam	Multiple_Ac	Add_To_Li																		
96	Brandon	Major	bmajor@ms	222-1219				true	true																		

Finishing Up

Your queries have been created and saved. What now? After saving your query you should be taken to the first page of the Query Manager tab, which now displays a list of queries you have created. It provides an overview of familiar information about your queries such as query name, and query conditions.

The screenshot shows the Query Manager interface with the following annotations:

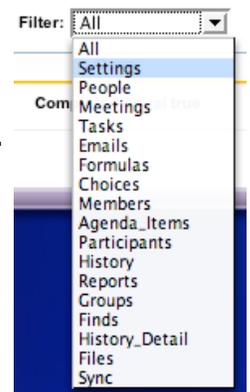
- Saved queries:** Points to the 'TPS Reports' query in the list.
- Filter list according to selection:** Points to the 'Filter: All' dropdown menu.
- Query position:** Points to the 'Position' column header.
- Query name:** Points to the 'Query Name' column header.
- Query:** Points to the 'Query' column header.
- Selection query is applied to:** Points to the 'Selection' column header.

There are some unfamiliar areas, however, that may need an introduction:

Filter: When your list of queries grows you may feel the need to filter them by selection.

By default the Filter is set to display queries from All selections.

To the right is a screen shot of selections for the 4D Meetings sample.



Query Position: More than one query means more than one tab in your selection window. You can set the position of your tabs here. A position of '1' means it will be the leftmost query tab in the selection window.

By following the settings in the screen shot below (using generic queries with the 4D Meetings demo):

Make sure the Filter is set to a specific Selection

Optionally display the All Records tab

Preset Queries

Filter: Display the All Records tab: [Create](#)

Position	Query Name	Query	Selection
<input type="text" value="1"/>	Query 1 <input type="checkbox"/>	Task_ID starts with 1	Tasks
<input type="text" value="2"/>	Query 2 <input type="checkbox"/>	Task_ID ends with 2	Tasks

Set Positions

Here is what we see when opening the Tasks selection window:

Task ID Name Assignee

Chapter 3: Using the Access Control Tab

Introduction

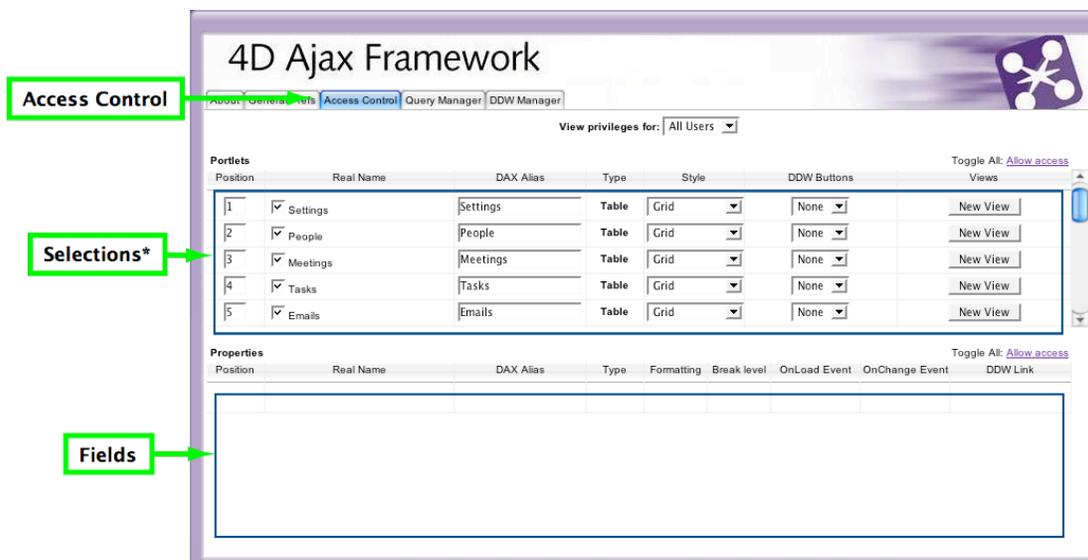
The Access Control tab is where you define privileges and customize the client environment for each User Group. As an administrator this is where a good chunk of your time can be spent. To get an overview of the various objects in this environment, please refer to *Chapter 1: Overview* in section *Control Panel -> Access Control*. To get your hands dirty with some functionality, read on.

Requirements

Before you begin, let's make sure everything is set with the 4D Ajax Framework.

- Install the 4D Ajax Framework to your database (see the "4D Ajax Framework Install" document).
- Log in as Administrator. (*Tip:* The 4D Ajax Framework defaults with login "Administrator" with no password. If you are rolling out your own password system then you must login with your own administrative account login and pass.)
- Click the Control Panel button.
- Click the Access Control tab.

The following is a screen shot of what you should see.



Congratulations on getting this far. Put your Administrator Hat on. It's time to get to work!

** As noted in Chapter 1, Selections do not just mean Tables. They also include Views and Developer Created Selections (DCS).*

Note: Screenshots are provided using the 4D Meetings Demo database. Feel free to open it to follow along.

Difficulty Meter

The Access Control tab is a centralized area in which a wide range of topics can be covered. To simplify things this chapter will be divided into two sections:

Getting Started: The Basics (★ or ★★)

Advanced Administrating (★★★ or ★★★★★)

★ = Least Difficult

★★★★ = Most Difficult

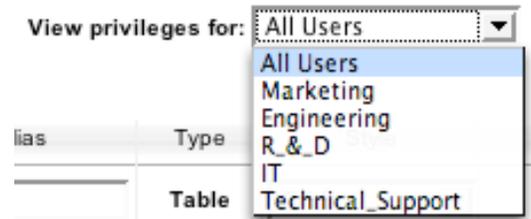
If you want to get familiar with the basic ins and outs of the Access Control tab go through the tasks covered in *Getting Started: The Basics*. Once you've mastered the basics head on over to *Advanced Administrating* for more dynamic customization.

Getting Started: The Basics (★ ★ ★ or ★ ★ ★ ★)

Use this section to get familiar with the Access Control tab. The following tasks are on the easier side of the difficulty scale, so if you are new to the 4D Ajax Framework this would be a good place to get your feet wet.

Choosing User Groups

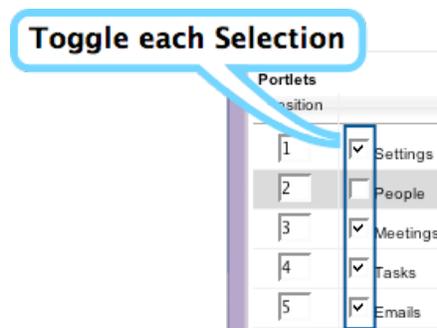
A good first step is to choose which User Group you would like to administer.



Near the top center of the window you will see a "View privileges for:" field. From the pull-down menu select the User Group you would like to define privileges for.

Toggling Access to Selections and Fields

You have the ability to show and hide which Selections each User Group has access to. In the Portlets area each Selection has a check box next to it. A *check* means the User Group has access to the Selection. An empty checkbox means that User Group will not have access.



√ = The Selection will be displayed to the User Group

You can also toggle all Selections On/Off at the same time by clicking the "Allow Access" link.



All Selections can be toggled on or off using the 'Allow access' link on the far right.



Enabling Fields is a bit different from enabling Selections. With Fields you have the option of making them available to the Output and/or Input form. To make a field available in the Output form make sure the checkbox to the right of 'O' is selected. To make a field available in the Input form make sure the checkbox to the right of the 'I' is selected.

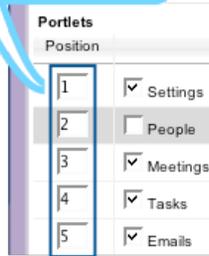


All Fields can be toggled on or off for either all Output or all Input forms.

Changing the Position of Selections and Fields

This section shows how to arrange the order of Selections in the Portal. At the far left of the Portlets area is the Position column. Rearrange the order at which your Selections are displayed by entering numerical values in this field. A value of '1' means that the Selection will be displayed at the top.

Change Position in Portal



Portlets	
Position	
1	<input checked="" type="checkbox"/> Settings
2	<input type="checkbox"/> People
3	<input checked="" type="checkbox"/> Meetings
4	<input checked="" type="checkbox"/> Tasks
5	<input checked="" type="checkbox"/> Emails

1 = Selection is at the top of the Portal

Field positions can be rearranged as well. First choose the Selection, and then you will be able to enter numerical values under the Position column for each Field. A value of '1' means that the Field will be the leftmost column in the Selection Window.

Change Field Positions



Properties			
Position	(O S)	Real Name	
1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	[People]First
2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	[People]Last
3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	[People]Age
4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	[People]City
5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	[People]State

1 = Field is the leftmost column

Renaming Selections and Fields

Selections can be renamed for the purpose of making them more understandable for your users. In the Portlets area is a column named DAX Alias. Customize what is in here to display more readable Selections to the user.



Portlets			
Position	Real Name	DAX Alias	Type
1	<input checked="" type="checkbox"/> Settings	Settings	Table
2	<input type="checkbox"/> People	People	Table
3	<input checked="" type="checkbox"/> Meetings	Meetings	Table
4	<input checked="" type="checkbox"/> Tasks	Tasks	Table
5	<input checked="" type="checkbox"/> Emails	Emails	Table

Field names can be customized as well. Choose your selection first, then you will be able to edit Field names under the DAX Alias column.

DAX Alias

Properties					
Position	(O S)	Real Name	DAX Alias	Type	
1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> [People]First	First	A
2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> [People]Last	Last	A
3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> [People]Age	Age	0 ^s
4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> [People]City	City	A
5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> [People]State	State	A

Formatting Fields

There are many formatting options available for your data fields. To customize formatting, go to the Formatting column of the Properties area.

Formatting

Toggle All: Allow access Q Allow access I				
Formatting	Break level	OnLoad Event	OnChange Event	DDW Link
None	None	<input type="checkbox"/>	<input type="checkbox"/>	4D Homepage
None	None	<input type="checkbox"/>	<input type="checkbox"/>	None
None	None	<input type="checkbox"/>	<input type="checkbox"/>	None
None	None	<input type="checkbox"/>	<input type="checkbox"/>	None
None	None	<input type="checkbox"/>	<input type="checkbox"/>	None

For each field select the appropriate format from the pull-down menu. Here are formatting examples for various data types:

Long Integer:

Long Int

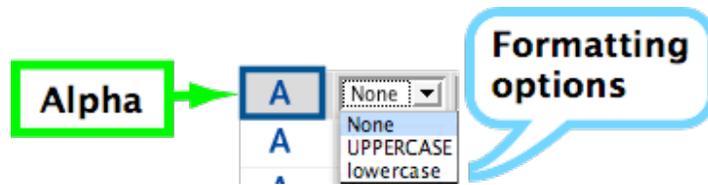
Long Int

23

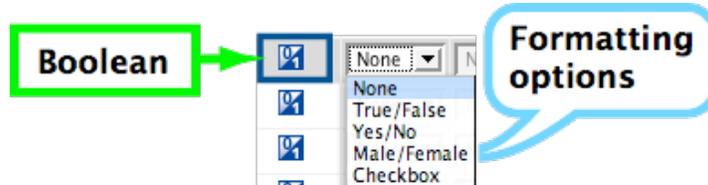
None
 9,999,999
 9,999.99 EUR
 \$9,999.99
 99.99 %

Formatting options

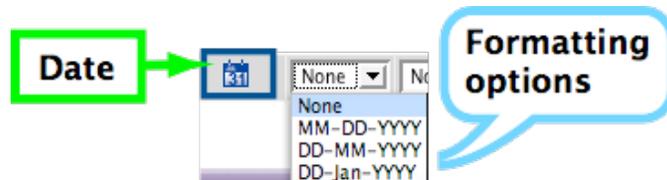
Alphanumeric:



Boolean:



Date:



Changing the Window Style

Selection windows can be viewed as a Data Grid, Calendar, Data Tree, Data Matrix, or Image Matrix.

Data Grid

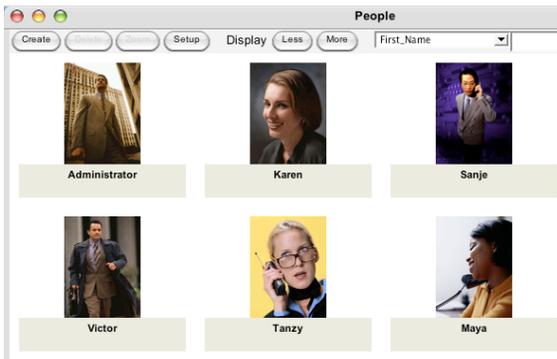
Group_ID	Name	Date_Created
33	Sales	11-16-2004
34	Marketing	11-16-2004
37	Management	11-18-2004
38	Customer Service	12-05-2004
32	Engineering	11-16-2004

DataTree

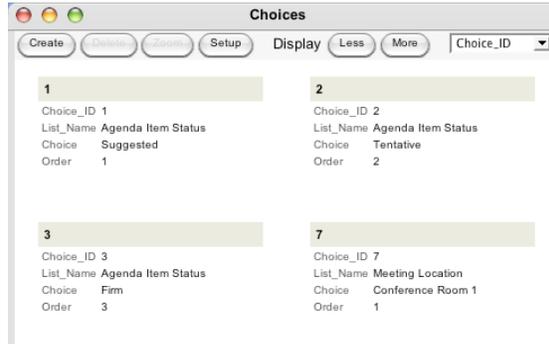
```

Delete
List
[-] Company Meeting (3)
    Purchase Margarita Mix
    Bring Pie
    Set up conference call
[-] Quarterly Staff Meeting (3)
    Bonus Distribution Poster
    Bring Wii Controllers
    TPS Reports
    
```

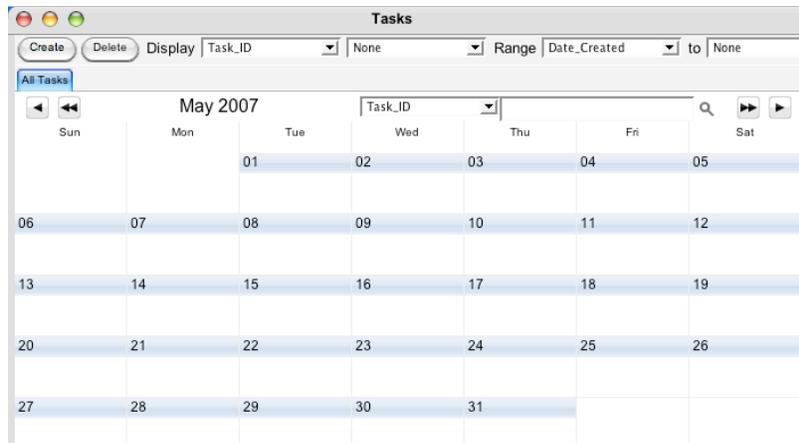
ImageMatrix



DataMatrix

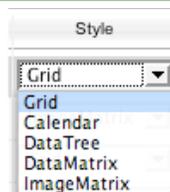


Calendar



To change window style, go to the Style Column of the Portlets area. Select any Style from the pull-down menu.

Choose Window Style



Advanced Administrating (★★★ or ★★★★★)

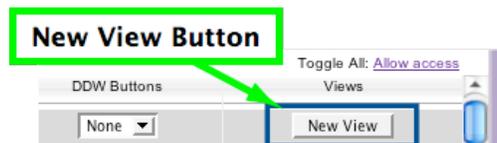
This section is for 4D Developers who demand more advanced customization.

Creating a View

Views can be thought of as virtual structures. They can be one-to-one copies of tables in your structure, or they can contain all fields from many tables in a relationship.

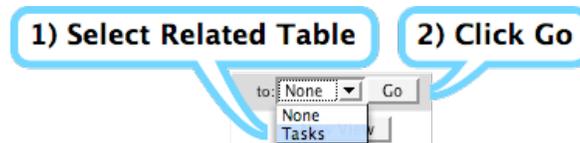
In the 4D Meetings Demo database, the Meetings table is the One table and the Tasks table is the Many table in a One-to-Many relationship.

To create a View you must first go to the One table in the relationship (in this case, the Meetings table). Click the 'New View' button.

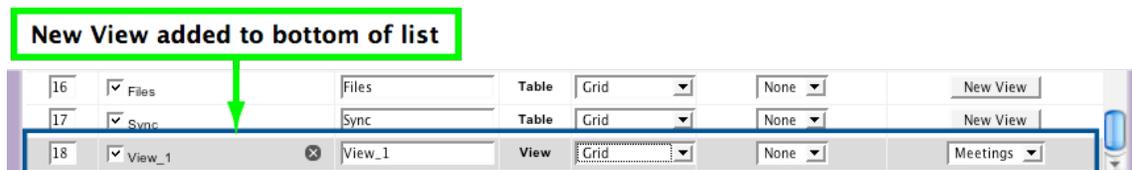


The current table selection (ie. Meetings table) can be thought of as a *starting point*. The table you select from the View pull-down menu (ie. Tasks table) can be thought of as an *ending point*. Any fields encompassed between the *starting* and *ending point* will be available in the new View.

Select the *ending point* from the pull-down list, then click the 'Go' button.



A new View will appear at the bottom of the list in the Portlets area.



The View can be modified like any other Selection in the Portlets area. Once you select it, you will see all the fields this virtual structure.

Below is a screenshot of the fields from a View in the 4D Meetings Demo database, where Meetings is the One table and Tasks is the Many table.

Fields are from Meetings and Tasks table

Position	Real Name	DAX Alias
16	[Meetings]Owner_ID	Owner_ID
17	[Meetings]Version	Version
18	[Tasks]Task_ID	Task_ID
19	[Tasks]Name	Name
20	[Tasks]Assigned_To	Assigned_To

Setting up Break Levels in Data Trees

Break Levels present information in a more straightforward manner. For further explanation of Break Levels please go to *Chapter 1: Overview* in section *Control Panel -> Access Control*.

First, choose the Selection you want to set to a Data Tree window style. In the 4D Meetings Demo database choose the Choices table.

Portlets	Position	Real Name	DAX Alias	Type	Style
	7	Choices	Choices	Table	DataTree
	8	Members	Members	Table	Grid
	9	Agenda_Items	Agenda_Items	Table	Calendar
	10	Participants	Participants	Table	DataTree
	11	History	History	Table	DataMatrix
					ImageMatrix
					Grid

Second, choose the Field to set to Break Level 1. In the demo database set field List_Name to Break Level 1.

Properties	Position	Real Name	DAX Alias	Type	Formatting	Break level
	1	[Choices]Choice_ID	Choice_ID	23	None	None
	2	[Choices]List_Name	List_Name	A	None	None
	3	[Choices]Choice	Choice	A	None	1
	4	[Choices]Order	Order	23	None	2
						3

Below is a screenshot to contrast life in the 4DAF client with and without Break Levels.

Life Before Break Levels

The screenshot shows a web application window titled "Choices" with a search bar and a table. The table has three columns: Choice_ID, List_Name, and Choice. The List_Name column contains repetitive values: "Agenda Item Status" (rows 1-3), "Meeting Location" (rows 7-9), and "Meeting Category" (rows 10-14). A blue box highlights the List_Name column.

Choice_ID	List_Name	Choice
1	Agenda Item Status	Suggested
2	Agenda Item Status	Tentative
3	Agenda Item Status	Firm
7	Meeting Location	Conference Room 1
8	Meeting Location	Conference Room 2
9	Meeting Location	Building 3
10	Meeting Category	Engineering
11	Meeting Category	Quality Control
12	Meeting Category	Sales
13	Meeting Category	Marketing
14	Meeting Category	Management

Data in List_Name is repetitive

Life After Break Levels

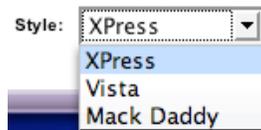
The screenshot shows a web application window titled "Choices" with a search bar and a table. The table has three columns: List, Choice_ID, and Choice. The List column contains distinct values: "Agenda Item Status (3)", "Meeting Category (6)", and "Meeting Location (3)". Each value has a drill down icon (a square with a minus sign and a list icon) to its left. The table lists specific choices for each list.

List	Choice_ID	Choice
Agenda Item Status (3)	1	Suggested
	2	Tentative
	3	Firm
Meeting Category (6)	10	Engineering
	11	Quality Control
	12	Sales
	13	Marketing
	14	Management
	15	All Departments
Meeting Location (3)	7	Conference Room 1
	8	Conference Room 2
	9	Building 3

Each distinct value in List_Name has a drill down icon

Unlocking More Themes

By default, three themes are loaded in the 4D Ajax Framework: XPress, Vista, and Mack Daddy.



To reveal more themes in the pull-down menu edit the file named index.html found in the dax folder (use index_fr.html for French version, index_es.html for Spanish version, or index_de.html for German version). Choose any HTML or text editor of your choice to edit the file.

You will see the following code in the <head> element:

```
<!-- stylesheets -->
<link rel="stylesheet" charset="ISO-8859-1" href="dax/themes/basic/basic.css"
media="all" type="text/css" title="XPress" />
<link rel="alternate stylesheet" charset="ISO-8859-1"
href="dax/themes/vista/vista.css" media="all" type="text/css" title="Vista" />
```

```
<link rel="alternate stylesheet" charset="ISO-8859-1"
href="dax/themes/osx/osx.css" media="all" type="text/css" title="Mack Daddy" />
<!--
<link rel="alternate stylesheet" charset="ISO-8859-1"
href="dax/themes/classic/classic.css" media="all" type="text/css"
title="McClassic" />
<link rel="alternate stylesheet" charset="ISO-8859-1"
href="dax/themes/gloss/gloss.css" media="all" type="text/css" title="Glossy" />
<link rel="alternate stylesheet" charset="ISO-8859-1"
href="dax/themes/newro/newro.css" media="all" type="text/css" title="Newro" />
<link rel="alternate stylesheet" charset="ISO-8859-1"
href="dax/themes/stationery/stationery.css" media="all" type="text/css"
title="Stationery" />
<link rel="alternate stylesheet" charset="ISO-8859-1"
href="dax/themes/twopointoh/twopointoh.css" media="all" type="text/css"
title="Two Point Oh!" />
<link rel="alternate stylesheet" charset="ISO-8859-1"
href="dax/themes/template/template.css" media="all" type="text/css"
title="Template" />
-->
```

This block of code contains a list of themes that are potentially available in the framework. The XPress, Vista, and Mack Daddy themes are listed at the top. The rest of the themes within `<!-- -->` are italicized which means they are commented out and unused.

Feel free to rearrange which themes you would like to dismiss or be made available. A commented out theme means that it will not be loaded in the framework.

Chapter 4: Developer Defined Windows and You

Introduction

Developer Defined Windows enable the 4D developer to display information in any way that they desire, so long as the content is HTML-based. In this chapter we will introduce the four basic types of DDWs as well as proof-of-concept examples on how to create them.

(Note: For an introduction into DDWs you can refer to section *DDW Manager* of *Chapter 1:Overview*.)

Requirements

Before you begin, let's make sure everything is set with the 4D Ajax Framework.

- Install the 4D Ajax Framework to your database (see the "4D Ajax Framework Install" document).
- Log in as Administrator. (*Tip:* For the 4D Meetings Demo database you can log in as 'Administrator' with no password.)
- Click the Control Panel button.
- Click the DDW Manager tab.

The following is a screen shot of what you should see.



Click the Create link. It's time to develop customizable windows!

5 Steps to Creating a DDW

Click the Create link, and then the following will be presented:

A screenshot of a web form for creating a DDW. The form has five numbered steps: 1. Title (text input), 2. Content (text input), 3. Type (dropdown menu showing 'New Window - Link'), 4. Associated to (dropdown menu showing 'Portal'), and 5. Create (button).

Here now are the five basic steps to creating a DDW:

1 Name the DDW.

Title

2 Define the content. Enter a URL or a 4D project method.

Content

3 Assign the Type. We will follow examples demonstrating each option.

Type
Associated to

4 Associate it to Portal or Other. We will follow examples demonstrating each option.

Associated to

5 Click the Create button.

Difficulty Meter

You may have noticed that many types of DDWs can be created based on the various options presented above. These variations, however, can be categorized into four general types of DDWs. Thus, four sections will follow and they are titled:

- *Static Link DDW* (★)
- *Dynamic Link DDW* (★★ ★)
- *HTML BLOB DDW* (★★ ★ ★)
- *Text Message DDW* (★★ ★)

★ = Least Difficult

★ ★ ★ ★ = Most Difficult

Each section will cover a typical example for each general type of DDW. Feel free to skip around from section to section based on your proficiency.

Note: Screenshots were taken using the 4D Tech Support Demo databases.

Static Link DDW (★)

A Static Link DDW is exactly what it sounds like. It is a DDW with its content from a static link. The path can either be absolute (<http://www.4d.com>) or relative (/list.html).

Here is an example of a static link DDW associated to the Portal. This type of DDW can be loaded like any other object from the Portal. Here are sample parameters when creating this type of DDW:

Title	<input type="text" value="4D Homepage"/>
Content	<input type="text" value="http://www.4d.com"/>
Type	<input type="text" value="New Window - Link"/>
Associated to	<input type="text" value="Portal"/> <input type="button" value="Create"/>

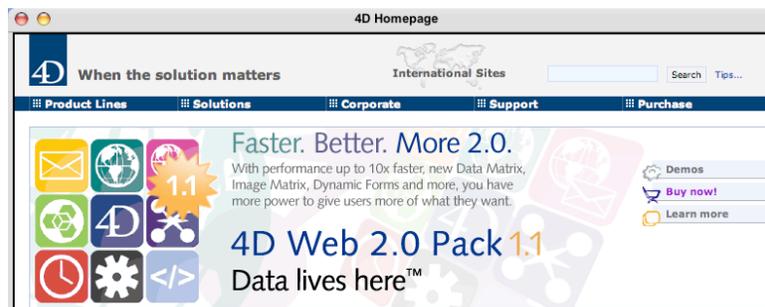
Notes:

- Be sure to include <http://> if entering a web URL.
- A new window will be loaded with its content from a static link.
- This ties the DDW to the Portal.

Click the Create button. The following object appears in the Portal.



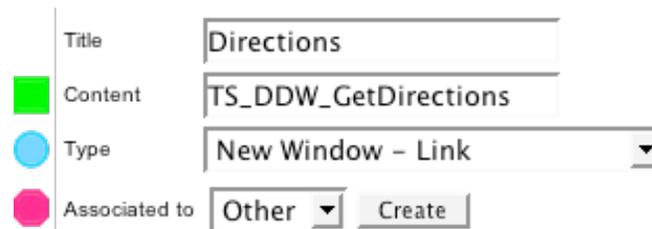
Load the DDW like you would any other object in the Portal. A new window opens and this is presented.



Dynamic Link DDW (★★★)

Dynamic Link DDWs load a dynamic HTML page or a dynamic link to another web site. The links are generated by your 4D project method. The method does all the work of harmoniously meshing data from the backend with the resulting dynamic URL. Thus, all effort must be focused in making sure the method does its job successfully.

Here is an example of creating a dynamic link DDW that will be associated to the field level (using the Tech Support Demo database):



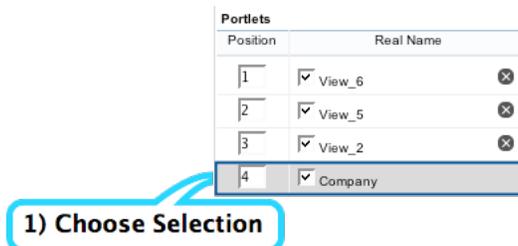
The screenshot shows a configuration form for a Dynamic Link DDW. It has four main sections: Title, Content, Type, and Associated to. The Title field contains 'Directions'. The Content field contains 'TS_DDW_GetDirections'. The Type field is a dropdown menu set to 'New Window - Link'. The Associated to field is a dropdown menu set to 'Other', with a 'Create' button next to it.

Notes:

-  Project method *TS_DDW_GetDirections* generates a Google Maps URL with address information of the currently selected record.
-  A new window will be loaded with its content from the dynamic link.
-  This enables the DDW to be tied to the field level.

Click the create button. Now it is time to associate the DDW to a field. Go to the Access Control tab. Choose the selection, then the field you would like the DDW to be assigned to.

In this example for the Tech Support Demo database, we chose the Company Table.



The screenshot shows a table titled 'Portlets' with two columns: 'Position' and 'Real Name'. There are four rows. The first three rows are 'View_6', 'View_5', and 'View_2'. The fourth row is 'Company'. A blue callout box with the text '1) Choose Selection' points to the 'Company' row.

Position	Real Name
1	<input checked="" type="checkbox"/> View_6 <input type="checkbox"/>
2	<input checked="" type="checkbox"/> View_5 <input type="checkbox"/>
3	<input checked="" type="checkbox"/> View_2 <input type="checkbox"/>
4	<input checked="" type="checkbox"/> Company <input type="checkbox"/>

Then the [Company]Name field.

Properties	
Position	Real Name
1	O: <input checked="" type="checkbox"/> I: <input checked="" type="checkbox"/> [Company]Company_ID
2	O: <input checked="" type="checkbox"/> I: <input checked="" type="checkbox"/> [Company]Name

2) Choose Field

Once the field is selected, assign the DDW. Go to the DDW Link column and select the DDW from the pull-down menu.

DDW Link
View Cases
Directions
None
Directions

3) Choose DDW

The DDW is now assigned to the field. Go to the Portal and load the selection that has the field with the assigned DDW.

Selection

Company

Create Delete Search Company_ID Save

All Company

Company_ID	Name	Address	Phone
1	Acme	797 E Julian St San Jose, CA 95112	4085551234

Field DDW is assigned to

DDW link

Click the DDW link. A new window with the dynamic link is loaded. For this example in the Tech Support Demo database, every time the user clicks the company name a new window loads with the company address in Google Maps (as well as directions from 4D San Jose Headquarters to the company location).

Acme

Web Images Video News Maps Gmail more Saved Locations Sign in Help

Google Maps 3031 Tisch Way, San Jose, CA 95128 797 E Julian St San Jose, CA 95112

Search the map Find businesses Get directions

Search Results My Maps Print Email Link to this page

Avoid highways Street View New Get reverse directions

From: 3031 Tisch Way San Jose, CA 95128

Drive: 7.6 mi (about 12 mins)

1. Head east on Tisch Way toward Dudley Ave

©2007 Google - Map data ©2007

For reference here is the code for project method *TS_DDW_GetDirections*:

```
C_TEXT($0)
```

```
ARRAY LONGINT($RecordNumbers_al;0)

DAX_Dev_DDW_GetAttributes ("RecordNumbers";->$RecordNumbers_al)

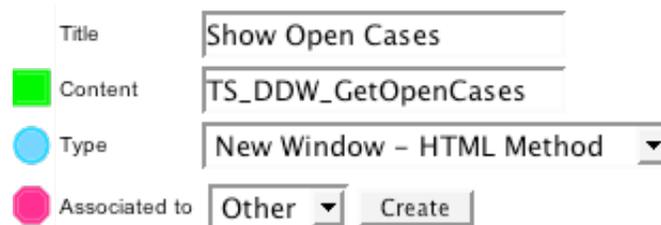
If (Size of array($RecordNumbers_al)=1)
    GOTO RECORD ([Company];$RecordNumbers_al{1})
Else
    UNLOAD RECORD ([Company])
End if

$0:="http://maps.google.com/maps?saddr=3031+Tisch+Way,+San+Jose,+CA+95128
&daddr="+[Company]Address
```

HTML BLOB DDW (★★★★)

HTML BLOB DDWs allow the developer to create and load BLOBS into the 4D Ajax Framework client environment. This type of DDW has the most potential in terms of customization since the resulting content can involve some Javascript code as well. Some 4D programmers may not be familiar with Javascript so this is why this type of DDW is weighed heavier on the difficulty rating scale.

Here is an example of creating an HTML BLOB DDW that will be associated to the selection window level (using the 4D Tech Support Demo database):



The screenshot shows a configuration form for a DDW. It has four rows, each with a colored icon on the left and a text input field on the right. The first row has a grey icon and the text 'Title' with the value 'Show Open Cases'. The second row has a green icon and the text 'Content' with the value 'TS_DDW_GetOpenCases'. The third row has a blue icon and the text 'Type' with a dropdown menu showing 'New Window - HTML Method'. The fourth row has a pink icon and the text 'Associated to' with a dropdown menu showing 'Other' and a 'Create' button to its right.

Notes:

-  Project method *TS_DDW_GetOpenCases* returns a BLOB.
-  A new window will be loaded with its content from the HTML method.
-  This enables the DDW to be tied to the selection level.

Click the create button. Now it is time to associate the DDW to a selection window. Go to the Access Control tab. Choose the selection you want to DDW to be associated to.

In this example with the 4D Tech Support database, we chose the Case List view.



Then go to the DDW Buttons column, and select the DDW from the pull-down menu. For this example we chose the DDW we created, Show Open Cases.

Position	Real Name	DAX Alias	Type	Style	DDW Buttons
1	<input checked="" type="checkbox"/> View_6	<input checked="" type="checkbox"/> Case List	View	Grid	Show Open Cases
2	<input checked="" type="checkbox"/> View_5	<input checked="" type="checkbox"/> Case Calendar	View	Calendar	None Directions Skype Tech Count Open Cases Get Actions View Cases Show Open Cases
3	<input checked="" type="checkbox"/> View_2	<input checked="" type="checkbox"/> Taken Actions	View	Grid	
4	<input checked="" type="checkbox"/> Company	Company	Table	Grid	
5	<input type="checkbox"/> Contact	Contact	Table	Grid	

2) Choose DDW

The DDW is now assigned to the selection. Go to the Portal and load the selection with the assigned DDW.



Click the DDW button. A new window with the HTML BLOB is loaded.

CASE ID	ENGINEER	DATE OPENED
1	Larry	4/23/07
3	Mike	4/25/07
4	Larry	3/20/07
6	Mike	4/26/07
9	Mike	4/30/07
14	Larry	5/1/07

For reference here is the code for *TS_DDW_GetOpenCases*.

```

C_BLOB($0;$htmlBlob_xb)

READ ONLY([Case])
QUERY([Case];[Case]Date_Closed=!00/00/00!)

DOCUMENT TO BLOB("open.html";$htmlBlob_xb)
PROCESS HTML TAGS($htmlBlob_xb;$htmlBlob_xb)

$0:=$htmlBlob_xb

```

You may notice the call to open.html in the code above. If you open open.html from the 4D Tech Support Demo database you will see Javascript code and calls to other 4D project methods, which more so customize the content of the DDW.

Further explanation of the Javascript code is beyond the scope of this manual. 4D developers are encouraged to seek other resources (such as daxipedia.4d.com) to learn how to integrate Javascript with 4D.

Text Message DDW (★★★)

Text Message DDWs can present customizable text messages based on data values in the database. They can be useful in performing calculations. They can also be creatively used to launch other applications (*see the Skype DDW in the 4D Tech Support demo database*).

Here is a typical example of creating a text message DDW that displays a custom text message based on queries and calculations from the backend (using the 4D Tech Support demo database):

Title: Count Open Cases
Content: TS_DDW_GetCaseCount
Type: Status Message - Text Method
Associated to: Other
Create

Notes:

- Project method *TS_DDW_GetCaseCount* returns the text message.
- The status message will be based on the text method.
- This enables the DDW to be tied to the selection window. Text message DDWs cannot be tied to the Portal.

Click the create button. Now it is time to associate the text message to a selection window. Go to the Access Control tab. Choose the selection you want to DDW to be associated to.

In this example with the 4D Tech Support database, we chose the Case Calendar view.

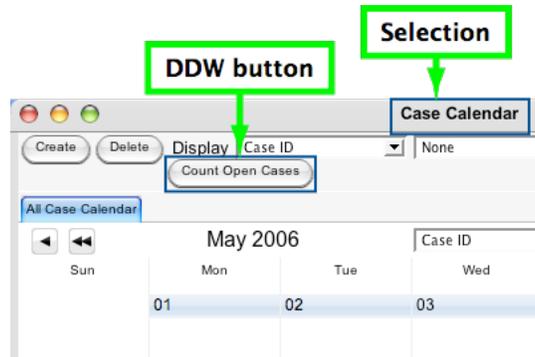


Then go to the DDW Buttons column, and select the DDW from the pull-down menu. For this example we chose the DDW we created, Count Open Cases.

Position	Real Name	DAX Alias	Type	Style	DDW Buttons
2	View_5	Case Calendar	View	Calendar	Count Open Cases
3	View_2	Taken Actions	View	Grid	None
4	Company	Company	Table	Grid	Directions
5	Contact	Contact	Table	Grid	Skype
6	View_1	Contacts	View	Grid	Tech

2) Choose DDW

The DDW is now assigned to the selection. Go to the Portal and load the selection with the assigned DDW.



Click the DDW button. A custom text message appears in the bottom border of the selection window.

2 Open, 5 Under Investigation

For reference here is the code for *TS_DDW_GetCaseCount*.

```

C_TEXT($0;$Result_t)
C_LONGINT($i;$Open_1;$UnderInvestigation_1)
ARRAY LONGINT($RecordNumbers_al;0)

QUERY([Case];[Case]Status="Open";*)
QUERY([Case]; | ;[Case]Status="Under Investigation")
FIRST RECORD([Case])
For ($i;1;Records in selection([Case]))
    Case of
        : ([Case]Status="Open")
            $Open_1:=$Open_1+1
        : ([Case]Status="Under Investigation")
            $UnderInvestigation_1:=$UnderInvestigation_1+1
    End case
NEXT RECORD([Case])
End for

$Result_t:=$Result_t+String($Open_1)+Char(Space)+"Open"+", "+Char(Space)
$Result_t:=$Result_t+String($UnderInvestigation_1)+Char(Space)+"Under
Investigation"
$0:=$Result_t

```

Associating DDWs

The example DDWs in the previous sections were associated to where we felt they would best apply (for example, a static link DDW to the Portal). However, each type of DDW can be assigned to other areas as well.

Here is a table showing the four types of DDWs with respect to the various areas they can be applied to. An "X" means that option is available.

Associate DDW to Portal or Other

	Static Link DDW	Dynamic Link DDW	HTML BLOB DDW	Text Message DDW
Portal	X	X	X	
Other	X	X	X	X

DDWs can be associated to three areas:

- 1) The Portal,
- 2) The Selection level, or
- 3) The Field level

This section of the chapter serves as a general reference for associating DDWs to other areas (if that option is available).

The Portal

Associating a DDW to the Portal involves 1 step:

- 1) Select 'Portal' in the Associated To field.



Once that is selected click the Create button. The DDW object then appears in the Portal. For example:



The DDW can now be loaded like any other object in the Portal.

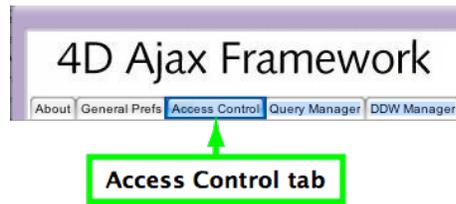
The Selection Level

Associating a DDW to the selection level involves 4 steps:

- 1) Select 'Other' in the Associated To field. Click the Create button.



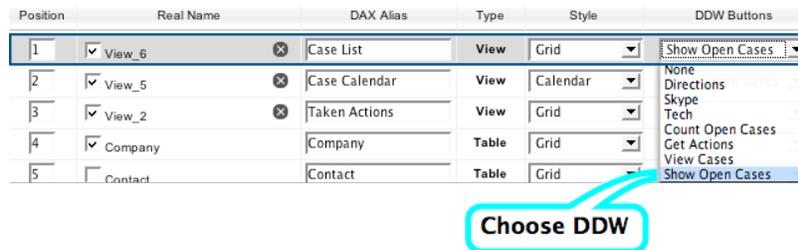
- 2) Go to the Access Control tab.



- 3) Choose the selection.



- 4) Choose the DDW from DDW Buttons pull-down menu.



The DDW is now associated to the selection. Load the selection from the Portal and click the DDW button to load the DDW.



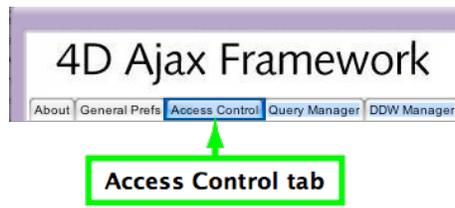
The Field Level

Associating a DDW to the field level involves 5 steps:

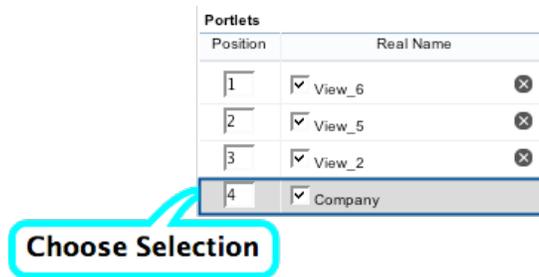
- 1) Select 'Other' in the Associated To field. Click the Create button.



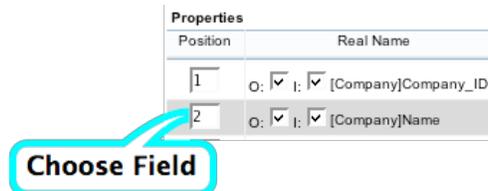
- 2) Go to the Access Control tab.



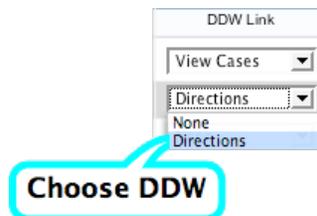
- 3) Choose the selection.



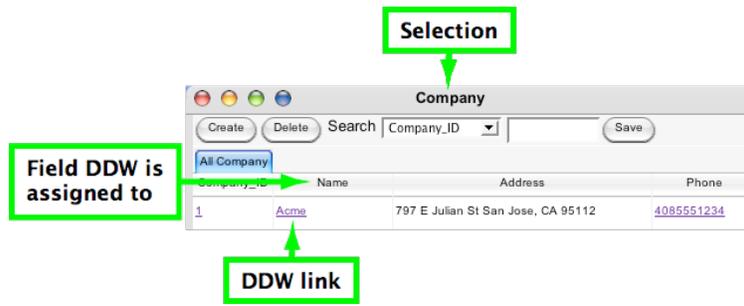
- 4) Choose the field



- 5) Choose the DDW from DDW Link pull-down menu.



The DDW is now associated to the field. Load the selection from the Portal and click the DDW link to load the DDW.



Chapter 5: Dashboards

Introduction

The 4D Ajax Framework v11 introduces a new toolkit named the Dashboard Editor for creating Dashboards for the web or the iPhone/iPod Touch. It has a host of features such as business indicators, charts, pixel-precise column resizing, integration with iPhone native applications, and more all bundled with a friendly user interface for creating Dashboards in a drag and drop manner.

This chapter provides a solid overview of Dashboards and the Dashboard Editor. For examples on building Dashboards please see “*iPhone Dashboard*” (*iPhone/iPod Touch*) and the “*Dashboard Handbook*” (*Desktop*). These documents come bundled with Demo databases to better aid you in learning how to build Dashboards.

iPhone Dashboards



Branch	Budget	Spent
2005		
Pine House	\$25,500.00	\$14,072.00
MGS Inc.	\$10,000.00	\$3,795.00
May Flower	\$13,000.00	\$13,392.00
Montgomery	\$19,500.00	\$22,820.00
Kingston	\$11,500.00	\$3,422.37
2006		
Pine House	\$23,000.00	\$5,346.00

iPhone Dashboards are data-at-a-glance summaries pulled from your 4D database. They provide quick access to critical data reports while on the go.

Desktop Dashboards

Symbol	Stock Name	Current	Trend
BGO	Bingo Inc	272.46	▼ -0.01%
EUT	Endlago & Track Inc	240.43	▼ -0.03%
GKU	Gondor King United	217.21	▼ -0.01%
LI	Liao Factories	26.42	▲ 0.27%
TAC	TransAmerica Cruises	18.01	▼ -0.32%
YYZ	Yuang Yeng Zaddle Inc	32.20	▼ -0.03%



Thus a tap of a cell in the *Spent* field switches to the *Percentage* field.

We can go back and forth as often as we wish. This is a great way to maximize use of the limited real estate on the iPhone.

Note: If a column has both Alternate Columns and an iPhone application applied to it, the Alternate Columns feature will take priority over the application.

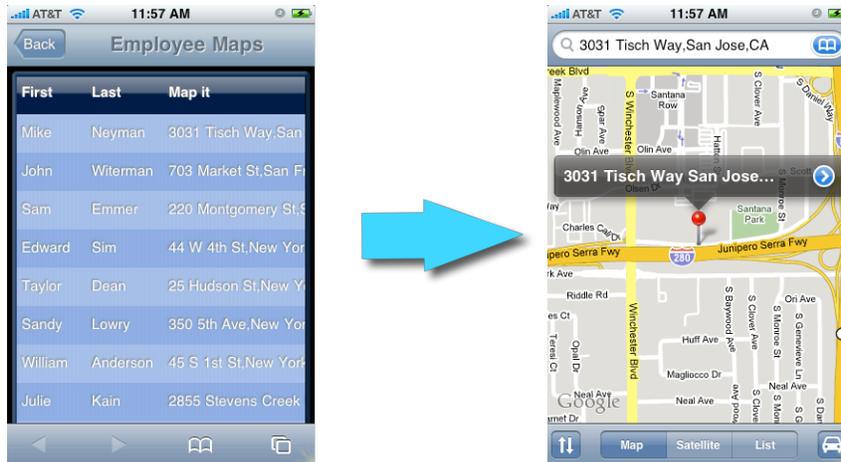
The order at which these conditions are listed is very important. The conditions can be thought of as Case statements. Thus, the first condition in the list that is evaluated as True exits the entire list and no longer evaluates the remaining conditions.

Integration with iPhone Applications

Dashboards can link directly with the following iPhone applications.

- Phone
- Email
- Maps (Google)
- Safari

Tap a field in the Dashboard, and data from that field can load these applications.



As we can see here, tapping an Address field can launch the Google Maps application.

Note: If a column has both Alternate Columns and an iPhone application applied to it, the Alternate Columns feature will take priority over the iPhone application.

Charts

iPhone Dashboard Charts use Developer Hook *Dax_DevHook_DefineChart* to define the values represented in the Chart, as well as the x-axis and y-axis text labels.



Desktop Dashboard Features

Indicators

Desktop Dashboards have the Indicators that are available to the iPhone, plus some Bonus Indicators as well.

The iPhone available Indicators are:

- Green
- Yellow
- Red
- Green button
- Yellow button
- Red button

The Bonus Indicators are not currently available in the drop-down menu of the Dashboard Editor. They are, however, available via Developer Hook *Dax_DevHook_SetIndicators*.

This is a list of the Bonus Indicators:



- Arrowup
- Arrowdown
- Noarrowyellow
-

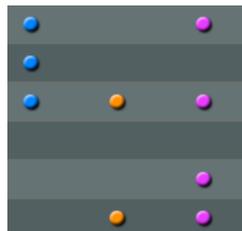


These Indicators work well if you want to display information with large font sizes. You can increase row height via CSS styling.



- Greenstrong
- Redstrong
- Green Triangle
- Red Triangle
- Yellow Square

These Indicators act as an alternative to the iPhone Dashboard Indicators.



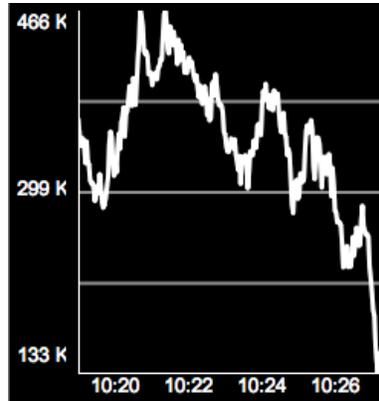
- Blue
- Orange
- Purple

This set of Indicators can work well for Boolean type fields.

Note: Indicators “Green button,” “Red button,” and “Yellow button” are fixed images. Thus they have fixed widths. If you extend your column width beyond the default width of 90 pixels, the text in your field may exceed the size of the image and so the information may look distorted.

Charts

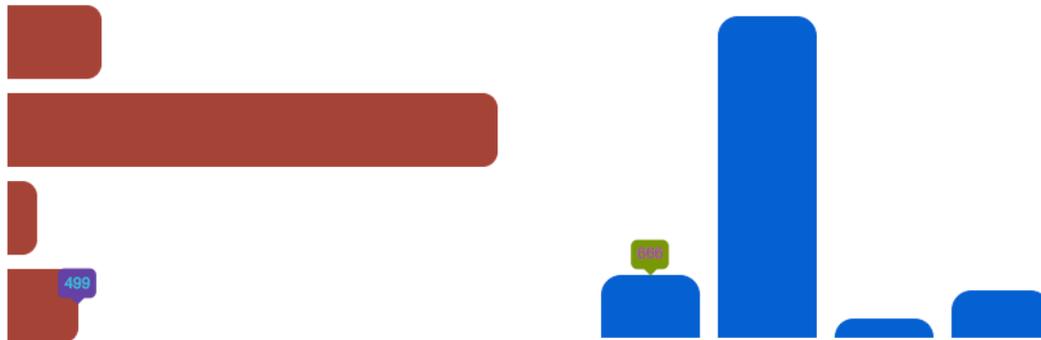
Desktop Dashboard Charts use Developer Hook *Dax_DevHook_DefineChart* to define the values represented in the Chart, as well as the x-axis and y-axis text labels.



Charts on the Desktop automatically resize to the spaces allotted to them on the HTML page.

Added in 4D Ajax Framework v11 Release 5 (11.5) are new Interactive Charts for the Desktop. Here is a quick look at them:

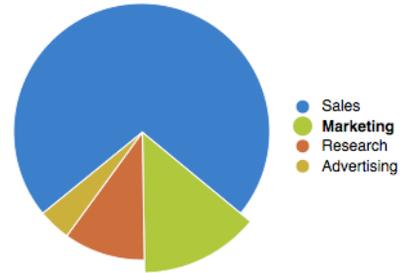
Bar Charts:



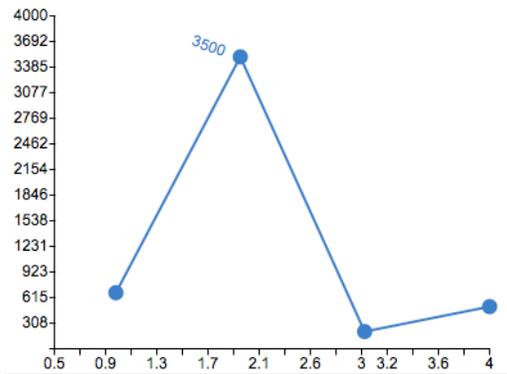
Dot Charts:



Pie Charts:

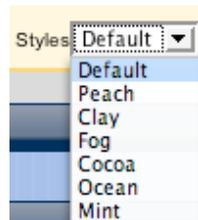


Line Charts:



Themes

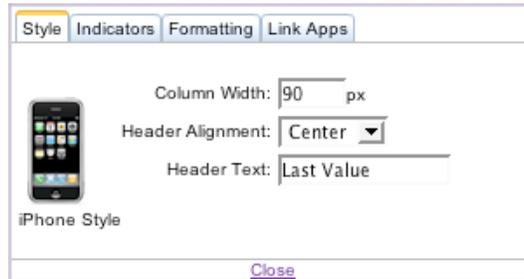
Desktop Dashboards have their own set of themes to choose from (as opposed to the Blue and Black themes for iPhone Dashboards).



Customization of headers and Detail Rows

We understand the importance of precision down to the very pixel when it comes to creating web pages. Thus, we have provided advanced

customization features such as column width (in pixels), header alignment, header text, and detail alignment for Dashboards.



Cross-Browser Compatibility

Desktop Dashboards have been built to look and function in the same manner across all major browsers – Safari 4+, Internet Explorer 7+, and Firefox 3+

The Dashboard Editor

The Dashboard Editor allows you to create Dashboards. To navigate to it:

- Log in to the 4D Ajax Framework Client (ex. <http://localhost:8080/index.html>)
- Hit the Control Panel button
- Go to the *Dashboard Manager* tab.
- Click the *Create*

This section will get you familiar with the Dashboard Editor. The same Editor is used to create Dashboards for the Desktop or for the iPhone/iPod Touch. Specify if you want to develop for the Desktop or the iPhone on the first page as seen below.

Dashboards [Cancel](#)

Create a Report for this selection: My Portfolio

Name: My Dashboard

Description: Desktop Dashboard

Platform Option: iPhone Support

[Start Dashboard Editor](#)

Here is the Dashboard Manager tab as part of the Control Panel in the 4D Ajax Framework.

- Choose the Selection (ie. Table, View, DCS) that the Dashboard will be based on.
- Give the Dashboard a name.
- Provide a description.
- Select 'iPhone Support' if you are developing for the iPhone. Leave the box unchecked otherwise.

Click the **Start Dashboard Editor** button to begin. You will then be presented with the following.

Dashboards

Drag the objects to the report:

▼ Database Objects

ID

BranchID

Name

► Formulas

Use Chart: Styles: Blue [Cancel](#) [Save](#)

▼ Add a group

ID

BranchID

Name

Click on a field to add a group:

This is the Dashboard Editor. Here you define what the Dashboard application will report on, and how it will be displayed. Let's take a look at the pieces that make up the Dashboard Editor.

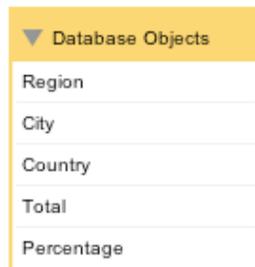
Editor



This provides a quick preview of what you are building. Drag fields to this area. Organize information by grouping it. Drag formulas to the Footer areas to perform calculations on specific columns. All these features and more can be done, and this editor outlines how the Dashboard will look.

Database Objects

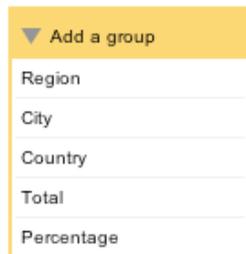
Drag the objects to the report:



These are the fields in your Selection. Drag and drop fields to the Editor area in the way you want them to appear in the Dashboard.

Add a group

Click on a field to add a group:



Group data in the Dashboard by clicking a field from this box.

Formulas



Drag Formulas to the Editor to perform calculations on field.

Charts



Toggle the ability to use Charts.

Show Alternate Columns (iPhone/iPod Touch only)



Use this feature to utilize Alternating Columns. Whenever an iPhone user taps on a column, the column will switch to a different column that you designate.

Style

Styles:



Choose among Blue or Black themed Dashboards for the iPhone.

For the desktop more themes are available:

- Peach
- Clay
- Fog
- Cocoa
- Ocean
- Mint

Property Inspector



Click this link to load the Property Inspector. If you then select a header or detail cell you can customize look and functionality. The Property Inspector loads with the following tabs:



- **Style:**
For a Desktop Dashboard it should display *Desktop Style*. Here you can alter text alignment for a particular detail cell. You can also customize column width, header alignment, and header text when you select a header cell.
- **Indicators:**
Set conditions for your Indicators here. Priority is given to the conditions listed at the top of the list.

The order at which these conditions are listed is very important. The conditions can be thought of as Case statements. Thus, the first condition in the list that is evaluated as True exits the entire list and no longer evaluates the remaining conditions.

- **Formatting:**
Select a style from the pull-down menu to format the field.
- **Link Apps:**
You can link applications to fields once they are clicked. You can link fields to the following:
 - a. Browser: Click a field that has a URL in it and it can load a webpage.
 - b. Mail: Click a field that has an email address in it and it can compose an email to that address using your default email client.
 - c. Maps: Click a field with an address in it and it can go to a Google Maps webpage displaying that address.

Note: Clicking on a Maps link will load the Google Maps page on the same window that contains the Dashboard. Thus, the page that contains the Dashboard will be exited.

Chapter 6: Data Grid

The new Data Grid introduced in 4D Ajax Framework v11 Release 1 (11.1) is a redesigned version of the existing Grid object built from the ground up. With a re-worked core the Data Grid is now a more universal object with a more granular API for many of the 4D Ajax Framework objects. It has a wealth of features such as the ability to:

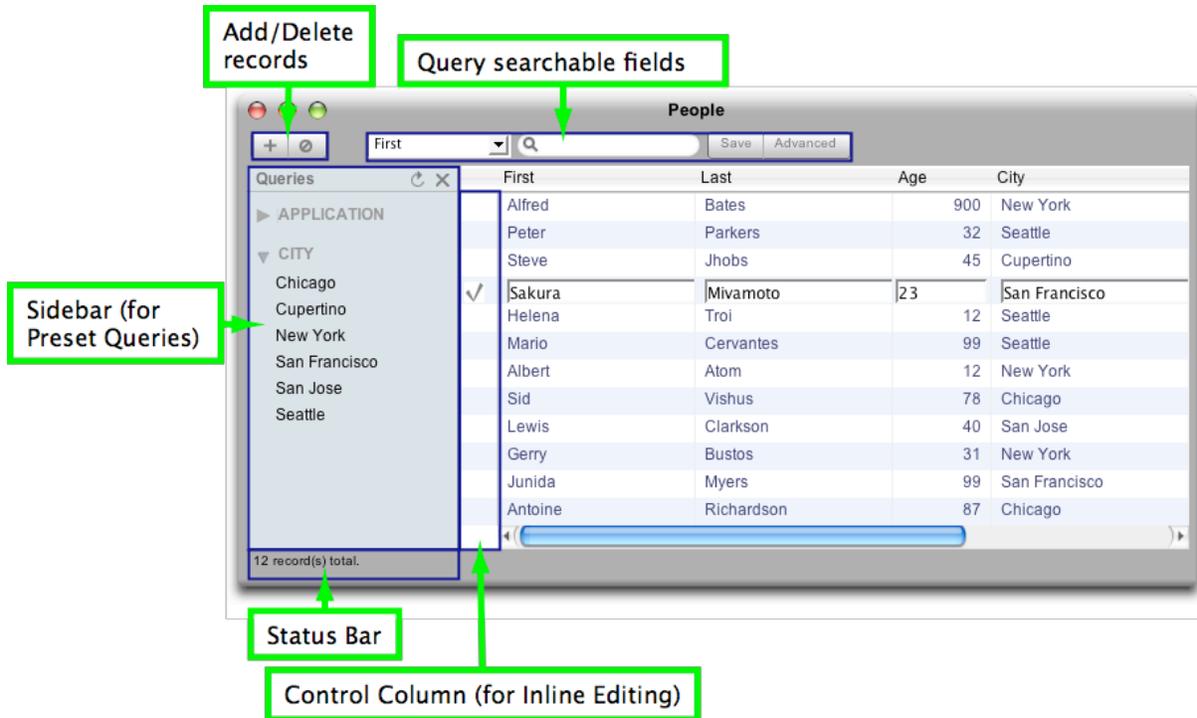
- Show and hide columns.
- Select single records, multiple records at the same time, or none at all.
- Lock columns, headers, and footers.
- Fire events when rows or cells are selected, and when data arrives from the back end.
- Apply CSS styling.
- And more...

	First	Last	Age	City
	Alfred	Bates	900	New York
	Peter	Parkers	32	Seattle
	Steve	Jhobs	45	Cupertino
✓	Sakura	Mivamoto	23	San Francisco
	Helena	Troi	12	Seattle
	Mario	Cervantes	99	Seattle
	Albert	Atom	12	New York
	Sid	Vishus	78	Chicago
	Lewis	Clarkson	40	San Jose
	Gerry	Bustos	31	New York
	Junida	Myers	99	San Francisco
	Antoine	Richardson	87	Chicago

12 record(s) total.

General Interface

Here is an overview of the Data Grid interface.



Add/Delete Records

Add or Delete selected records using the buttons at the top left corner as seen above.

 - Click this button to add a new record.

 - Click this button to delete selected records.

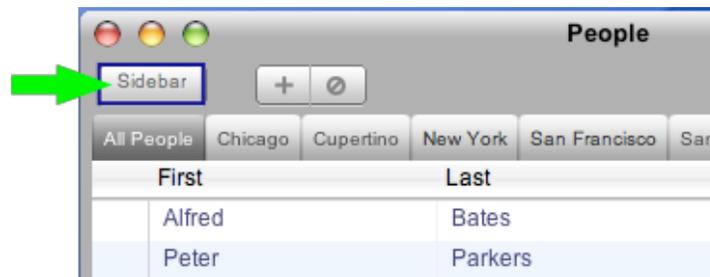
The Data Grid is defaulted to run in Multi-Selection mode. This means that users can select multiple records at a time. Users must use the CTRL key (PC) or the CMD key (Mac) to add or remove records from the Selection.

Preset Queries

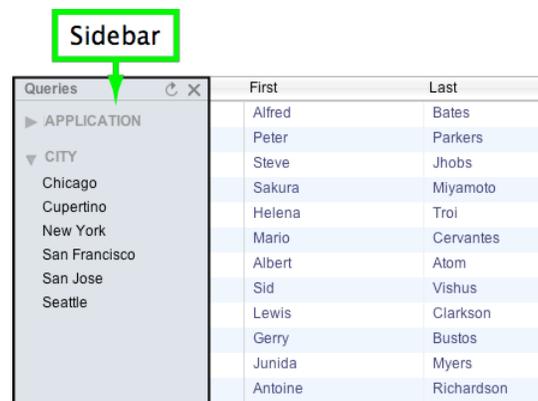
Preset Queries are created in the Query Manager tab of the Control Panel. Administrators create these queries ahead of time to make things convenient for the end user. Preset Queries can be displayed in the **Sidebar** or as **Tabs**.

Sidebar

The Sidebar is an area on the Data Grid displaying the Preset Queries as a list.



Hit the Sidebar button to display it.



This is what the Sidebar looks like. This example displays data driven Preset Queries in the Sidebar for the *City* field.

Tabs

Alternatively, Preset Queries can be represented as Tabs.

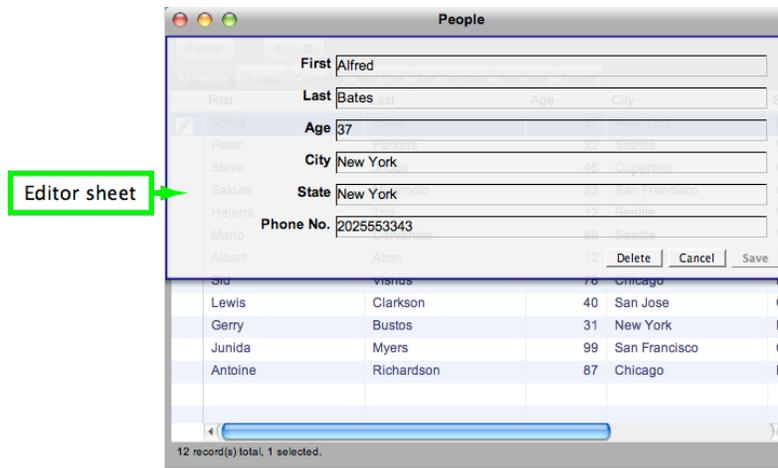


Hit the Tabs button to display them.



Here is a look at tabs. This example displays data driven Preset Queries as tabs for the City field.

Inline Editing

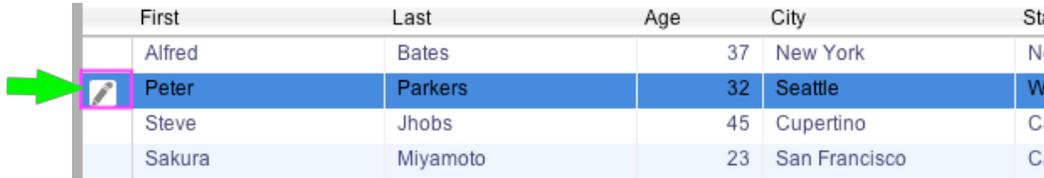


Double-click a record and an Editor sheet will appear.

Inline Editing, however, allows you to edit records in the Data Grid itself, not in another form.



Click within the Control Column of a record. A Checkmark will appear, and its fields will become editable.



First	Last	Age	City	State
Alfred	Bates	37	New York	N
Peter	Parkers	32	Seattle	W
Steve	Jhobs	45	Cupertino	C
Sakura	Miyamoto	23	San Francisco	C

Alternatively, first highlight a record. A Pencil icon will appear in the Control Column. This tells you that the record can now be edited for Inline Editing.

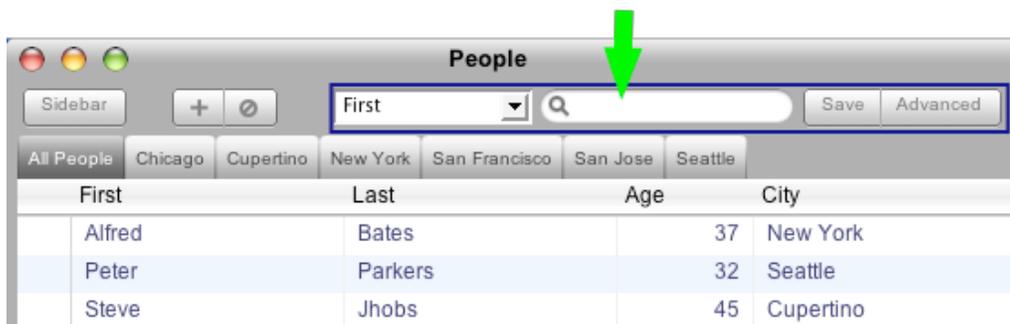
Next click the Pencil icon and then the fields will become editable.

Features

The Inline Editor supports many features:

- Choice Lists: Fields that are assigned Choice Lists in 4D automatically behave in the same way in the Data Grid.
- Keyboard Shortcuts:
 - Hit the *Esc* key to Cancel editing a record.
 - Hit the *Tab* key to go to the next field.
 - Hit the *Return* key to Save the record.
- A Calendar Picker appears for Date fields.
- Boolean fields appear as check boxes. The values depend on the formatting rules selected in the Control Panel.
- Callbacks: 4D Callbacks are supported during Inline Editing.

Search



The search bar appears when fields are designated as searchable.

Properties

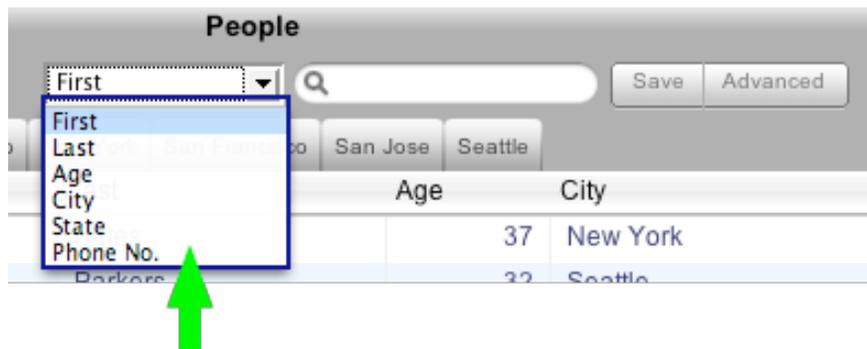
Position	(O S)	Real Name	DAX Alias
1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> People]First	First
2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> People]Last	Last
3	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> People]Age	Age
4	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> People]City	City
5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/> People]State	State



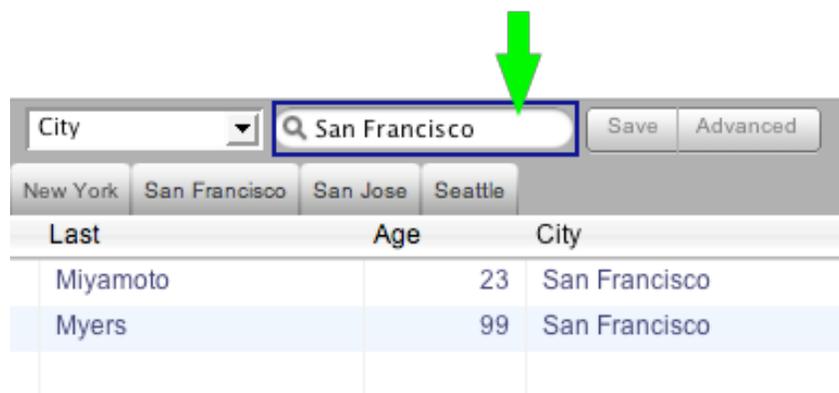
To make a field searchable go to the Control Panel -> Access Control tab. Choose the appropriate selection. For each field, toggle the 'S' checkbox. A check means that field is searchable.

As seen in the above screenshot, all of the fields with the third checkbox (from the left) selected are searchable.

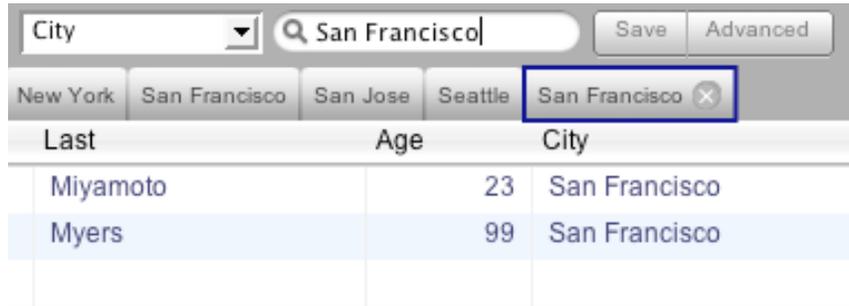
Basic Search Interface



The first pull-down menu displays searchable fields. Choose one to perform a search on it.



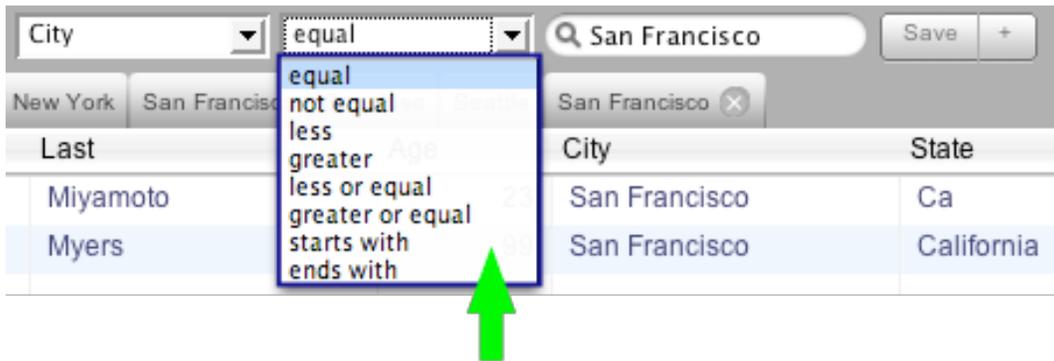
Enter the query condition in the search box. For text fields the Data Grid performs a 'starts with' query. For numeric fields the Data Grid performs an 'equal to' query.



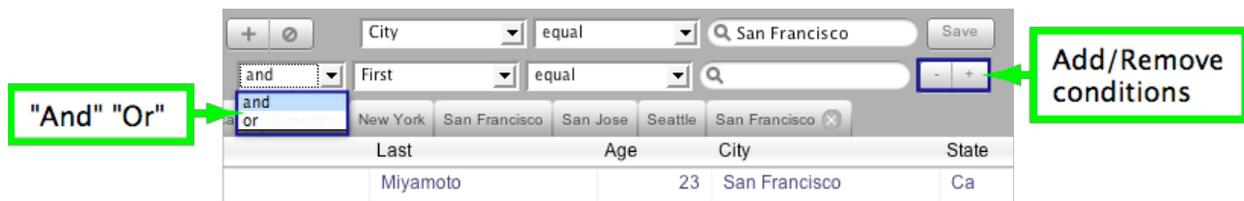
Hit the Save button and the query appears as a tab, much like a Preset Query tab. Unlike a Preset Query tab, however, this one is user-defined. It can be deleted by hitting the 'x' icon within the tab. Also, this tab is not preserved if the user logs out and logs back in to the framework.

Advanced Search Interface

Hit the 'Advanced' button, and the search interface will change a bit.



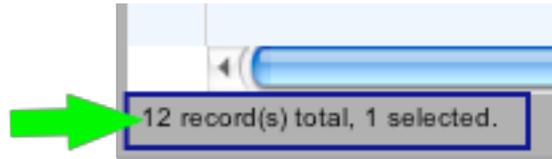
A new pull-down menu appears. It reveals query operators that add further customization for the user queries.



Hit the '+' button (far right) and a new line for another query condition appears. Combine this new condition with the previous one by choosing "And" or "Or".

Add more conditions as you see fit. However, adding too many conditions may compromise the appearance of the Data Grid.

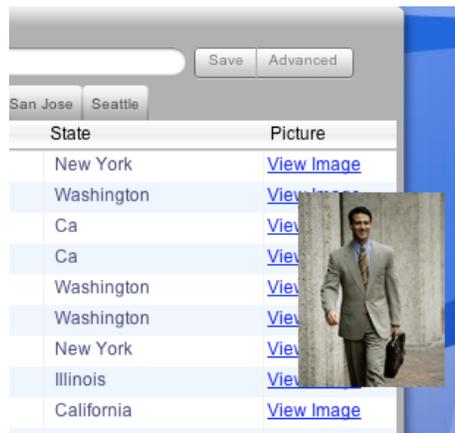
Status Bar



The Status Bar displays the total number of records and the number of records currently selected.

Picture Preview

Hover over the 'View Image' link of a Picture field and a preview of that image appears.



Chapter 7: Offline Mode

Imagine connecting to your 4D Ajax Framework (4DAF) powered webpage in your office and then deciding to go to the factory to gather more data. There's a good chance you may lose your laptop's online connection between the office and factory. How can you gather and edit new information if communication with your database is lost?

The solution is Offline Mode, a brand new feature implemented in the 4DAF Release 2 v11.2. Set your web application to Offline Mode and you can now travel anywhere to save your records locally and then upload them to 4D at a later time.

Requirements

Web browsers that are HTML 5 compatible or have Google Gears installed support Offline Mode. Safari 3 and Firefox 3 support HTML 5. Users of Internet Explorer 7 and Firefox 2+ must have Google Gears installed to use Offline Mode.

As of Release 2 v11.2 of the 4DAF, HTML 5 is the latest revision of the core language of the World Wide Web. Offline storage is a new native feature of HTML 5, which is why web browsers that are compatible with HTML 5 automatically support Offline Mode. For more information on HTML 5, see http://en.wikipedia.org/wiki/HTML_5.

At this time Internet Explorer is not compatible with HTML 5. Thus, IE 7 users must install Google Gears, which is an open source implementation of HTML 5. Google Gears can be downloaded at <http://gears.google.com/>.

Currently, only the 4DAF's Data Grid object supports Offline Mode. Offline records can be added, modified, and erased via both Inline Editing and the sheet window editor.

Currently, picture fields are not supported in Offline Mode.

Offline Mode is not a Full Client

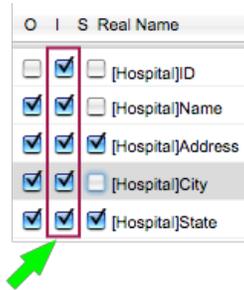
A common misconception that arises with the mention of Offline Mode is that it must equate to a full client in the web browser. This is not true. It does not allow for storage and caching of existing data offline. It can only add and modify records entered while offline.

Limitation

A noteworthy limitation of Offline Mode is that offline tables will not work correctly unless all Input fields for that offline table are set to viewable. To set all fields for viewable:

- 1) Log in to the 4DAF Client as Administrator.
- 2) Open the Control Panel. Go to the Access Control tab.

- 3) Select the table you intend to set for Offline Mode.
- 4) Go to that table's fields. Under the 'I' column, check all boxes (as shown below).

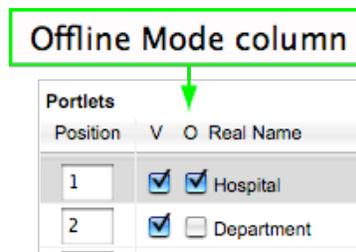


Using Offline Mode

Administrator: Enable Tables for Offline Mode

Administrators can log in to the 4DAF Client (for example, at <http://localhost:8080/index.html>) and can determine which tables will be cached for offline storage.

- 1) Log in to the 4DAF Client as Administrator.
- 2) Launch the Control Panel -> Access Control tab.
- 3) Mark a table's checkbox in the Offline Mode column to cache it for offline storage.



- 4) Then you must log out and log back into the 4DAF Client to see the Offline/Offline button in the Taskbar.



The Online/Offline button

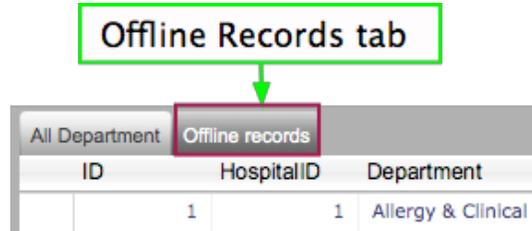
If no tables are set for offline storage, then the button is not displayed in the 4DAF Client.

End User: Going Offline and Back

The 4DAF Client is setup for Offline Mode from the get-go. Hit the Offline Mode button to cache required structure information and to place the client into Offline Mode.



Even though the connection with the server is not available, the end user can create and modify offline records as needed. An 'Offline Records' tab appears for tables that were designated for offline storage (To set a table for offline storage, see section "Administrator: Enable Tables for Offline Mode").

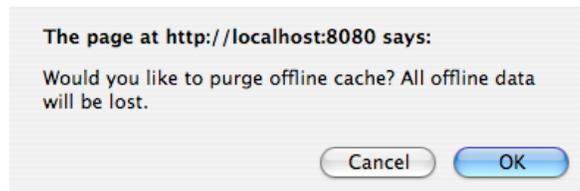


When the user returns to their original online location they can hit the Online button to upload all new and modified records to the 4D server.



Purge Offline Cache

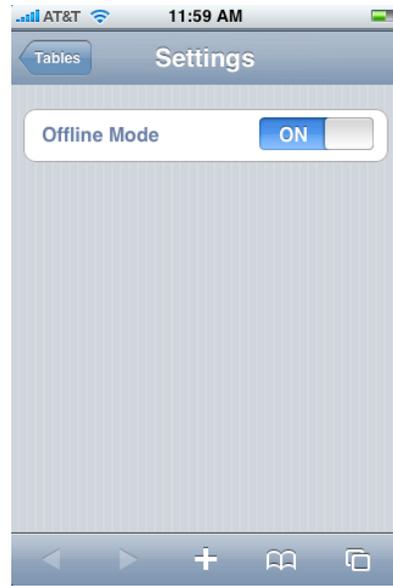
When toggling between Offline and Online Mode in the 4DAF Client you are provided with the option to clear the offline cache.



Click 'Ok' to clear the cache or 'Cancel' to deny the purge.

End User: Offline on the iPhone

The 4DAF on the iPhone supports Offline Mode as well. Once you log into the iPhone Client, you will see a 'Settings' button that appears at the screen that displays the list of Tables. Hit the 'Settings' button to toggle between Offline and Online Mode.



Chapter 8: iPhone Data Entry

The 4D Ajax Framework (4DAF) development team has heeded many of the requests of 4D Developers to push the limits of 4DAF on the iPhone even further. As a result, the 4D Ajax Framework Release 2 (v11.2) introduces Data Entry on the iPhone.

Here is an overview of its major capabilities:

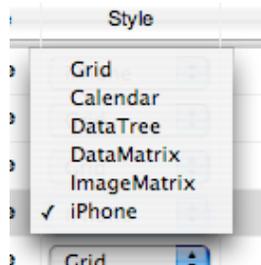
- Add or modify existing records
- Optimized for iPhone data entry and editing
- Look and feel of the native iPhone user interface
- Text fields editable using iPhone's native keyboard
- Dates are edited via iPhone's wheel rolodex
- Choice List and Boolean support
- And more...

To use this web application connect to `ipe.html` in the iPhone's Safari browser, for example <http://localhost:8080/ipe.html>.

Enable a table for iPhone

To make a table available on the iPhone log into the 4DAF Client as Administrator.

- 1) Go to the Control Panel -> Access Control tab.
- 2) From there go to the Style column and set it to 'iPhone' for any table you would like available on the iPhone.

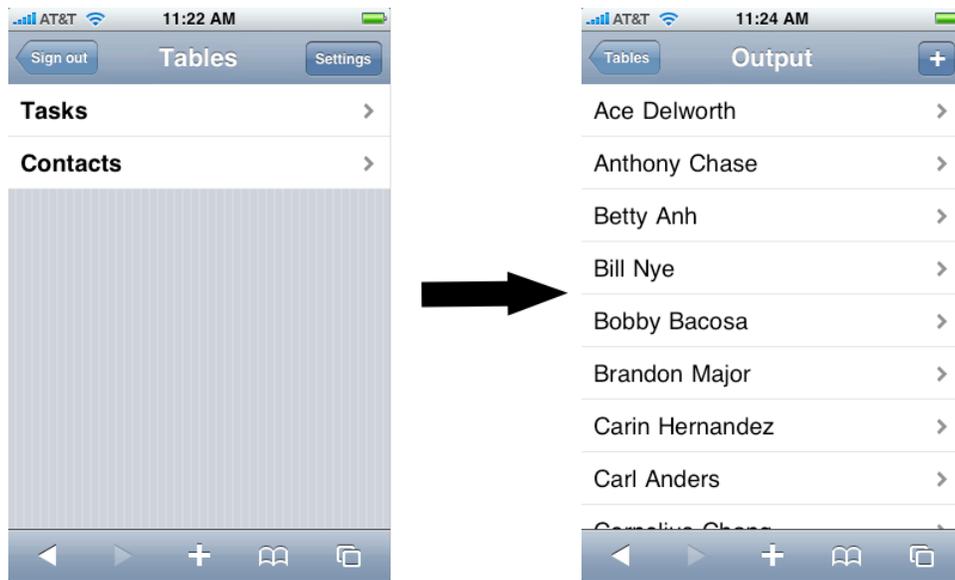


- 3) You can now connect to the web server via your iPhone. The list of available tables will appear after you log in.

The Output Form

After logging in, users are taken to the list of tables available for entry on the iPhone. Once they select a table the records are represented by the value of a single field.

In the example below, users select the 'Contacts' table and then see the actual contacts represented by the full name field.



This is different from a typical Output Form where the list can contain many fields for each record. Because of the limited real estate on the iPhone only a single field can represent a record in the Output form.

Choose the Field

To determine which field represents an iPhone record, make sure it has a value of '1' in the Position column. To change this value:

- 1) Go to the Control Panel -> Access Control tab.
- 2) Select the table.
- 3) Go to the Position column of the field you would like to represent the iPhone records, and set it to '1'.

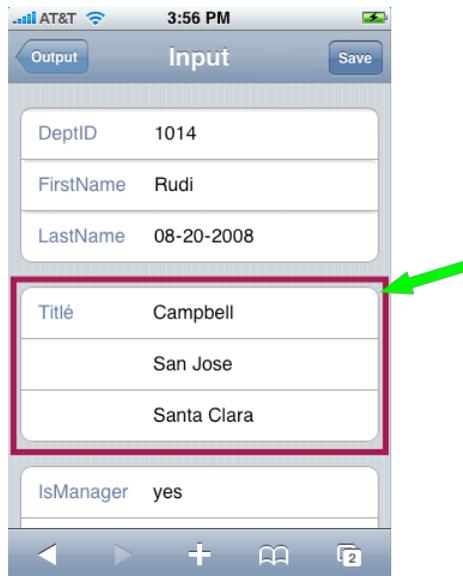
Choice Lists

Choice lists can be displayed in one of two ways:

- As a short list
- As a full choice list

Short list

If a choice list contains 3 or fewer values, all of the list values will display on the Input Form as seen below. Tap on any value to select it.



Full Choice List

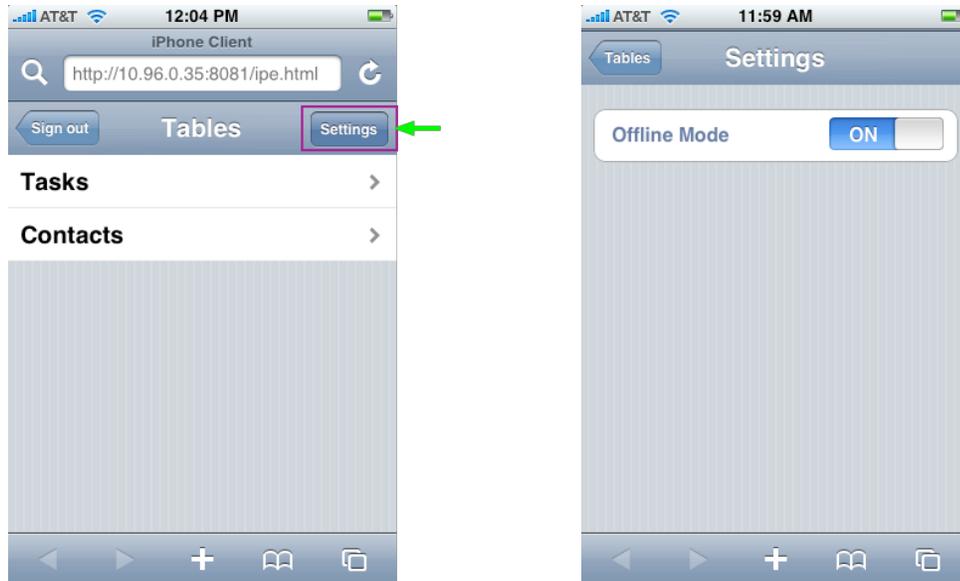
If a choice list contains 4 or more values, users will see the current choice value and an arrow indicating more options are available. Tap the choice list to bring up another screen displaying all choice list values.



Offline Mode on the iPhone

Choosing Offline Mode on iPhone

The 4DAF on the iPhone supports Offline Mode as well. Once you log into the iPhone Client, you will see a 'Settings' button that appears at the screen that displays the list of Tables. Hit the 'Settings' button to toggle between Offline and Online Mode.



To determine which tables are available for Offline Mode, see section “**Administrator: Enable Tables for Offline Mode**” in Chapter 7: Offline Mode.

Limitation

A noteworthy limitation of Offline Mode is that offline tables will not work correctly unless all Input fields for that offline table are set to viewable.