

Re: Record navigation with Paging

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.adonet/2005-04/msg00555.html>

- *From:* "Hari" <anonymous@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Mon, 11 Apr 2005 10:45:03 -0700
-

Thanks for the info Ryan,

The main thing is i don't want to keep more than 5 pages of data (5*30) in the memory. Also the main focus is user should be able to navigate with single record, in addition to a page navigation.

Also in my application the UI is build with unmanaged MFC app and only the storage access layer is in the managed DLL with ADO.NET. So i would like to build the custom DataView to accomodate these requirement.

Also the Table need to maintain a fixed number of records. (only last n(20,000) rows should be remain in the table).

Any more thoughts??

-Hari

>-----Original Message-----

>I'd use a DataTable, build a dataview on it, store it in session state, and

>then just set the RowFilter property

><http://www.knowdotnet.com/articles/advancedrowfilter.html>

to filter the

>data. 20,000 rows is a buttload of data and it's doubtful that you need that

>much data at one time - if you can use 5,000 and cache it, then if they

>select something in the next 5,000 hit the db again, and the same for the

>next 5,000 but assuming you do, you don't want to make unnecessary trips

>back to the db each time. If the user erroneously clicks forward then

>backward, you don't want to make two trips to the db on an accident. A

>DataView is probably the more efficient way to go in this scenario but as

Re: Record navigation with Paging

>with anything, the devil will be in the details.
>
>—
>W.G. Ryan MVP (Windows Embedded)
>
>TiBA Solutions
>www.tibasolutions.com | www.devbuzz.com |
>www.knowdotnet.com
>"Hari" <anonymous@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in
>message
>[news:191901c53ca9\\$5d2b3720\\$a601280a@xxxxxxxxxxx](mailto:news:191901c53ca9$5d2b3720$a601280a@xxxxxxxxxxx)
>> Hi,
>>
>> Iam working with a custom grid control to navigate
>> navigate large number of records from a table.
>> The control need to have row up, row down, page up,
>> page
>> down features.
>>
>> Requirements are :
>> page size = 30 records
>> maximum records in database table – 20,000 (each
>> record
>> 150K)
>>
>> I was looking the msdn sample:(Paging through a query
>> result) its shows building dynamic query for every
>> pages.
>> So inorder to this way, if i want to navigate a singe
>> row
>> (up/down (one record) do i need to build a query every
>> time? or is there any better way to approach this??
>>
>> please help me guys.....
>>
>> Hari
>>
>> The sample i was looking :
>> case "Next":
>> selCmd.CommandText = "SELECT TOP " + pageSize
>> + "
>> CustomerID, CompanyName FROM Customers " +
>> "WHERE CustomerID > @CustomerId
>> ORDER BY CustomerID";
>> selCmd.Parameters.Add("@CustomerId",
>> SqlDbType.VarChar, 5).Value = lastVisibleCustomer;
>> break;
>> case "Previous":
>> selCmd.CommandText = "SELECT TOP " + pageSize
>> + "
>> CustomerID, CompanyName FROM Customers " +

Re: Record navigation with Paging

```
>> "WHERE CustomerID < @CustomerId
>> ORDER BY CustomerID DESC";
>> selCmd.Parameters.Add("@CustomerId",
>> SqlDbType.VarChar, 5).Value = firstVisibleCustomer;
>> break;
>> default:
>> selCmd.CommandText = "SELECT TOP " + pageSize
+ "
>> CustomerID, CompanyName FROM Customers ORDER BY
>> CustomerID";
>>
>> // Determine total pages.
>> SqlCommand totCMD = new SqlCommand("SELECT
Count
>> (*) FROM Customers", nwindConn);
>> nwindConn.Open();
>> int totalRecords = (int)totCMD.ExecuteScalar();
>> nwindConn.Close();
>> totalPages = (int)Math.Ceiling((double)
>> totalRecords / pageSize);
>>
>> break;
>>
>>
>>
>
>
>.
>
.
```

• *References:*

- ◆ ***Record navigation with Paging***
 - ◇ *From:* Hari
- ◆ ***Re: Record navigation with Paging***
 - ◇ *From:* W.G. Ryan eMVP
- Prev by Date: ***Re: Record navigation with Paging***
- Next by Date: ***Re: adVarChar***
- Previous by thread: ***Re: Record navigation with Paging***
- Next by thread: ***Re: Record navigation with Paging***
- Index(es):
 - ◆ ***Date***
 - ◆ ***Thread***