Application Aspect

Hierarchical DataGridView v1.7
BOOKLET

Flexibility. Simplicity. Creativity." Our business philosophy relies on these three principles.

We believe these three ideas influence each other and give a deeper sense of purpose to our work which is to build easy to use, smart and flexibile controls for .NET Framework. Our goal is to make your job easier and shorter. We choose .NET Framework because we believe this technology is the best way to build fast and stable applications.

Hierarchical DataGridView is a brand new Windows Forms component that extends DataGridView component included in Microsoft .NET Framework 2.0 and .Net Framework 3.5. It is a unique grid control that organizes, edits and displays data in hierarchical form.

Hierarchical DataGridView can be seen as a mixture of Visual Studio 2005 or Visual Studio 2008 DataGridView, Microsoft Project or Microsoft Access data table and simple Excel sheet.

If you are a Visual Studio 2005 or Visual Studio 2008 developer and the terms such as data relation, self-referencing hierarchy, data structure, multilevel data, master-detail, parent-child, category-subcategory, tree or recursion describe the problem that you have, Hierarchical DataGridView is the right component to solve it.

Hierarchical DataGridView control supports data binding to any data source that implements IList, IListSource, IBindingList and IBindingListView interfaces such as BindingSource, Collections, Lists, Business Objects, etc. Binding to Nullable Data Types are also supported.

You can use Hierarchical DataGridView component in many various situation such as: project management, organizational structures, task management, file and directory browsing, etc.

OVERVIEW

How does Hierarchical DataGridView work?

After setting DataSource property and some more additional properties (PrimaryKeyColumnName, HierarchicalColumnName and RootValue), control is automatically displaying data in hierarchical form in the following way:

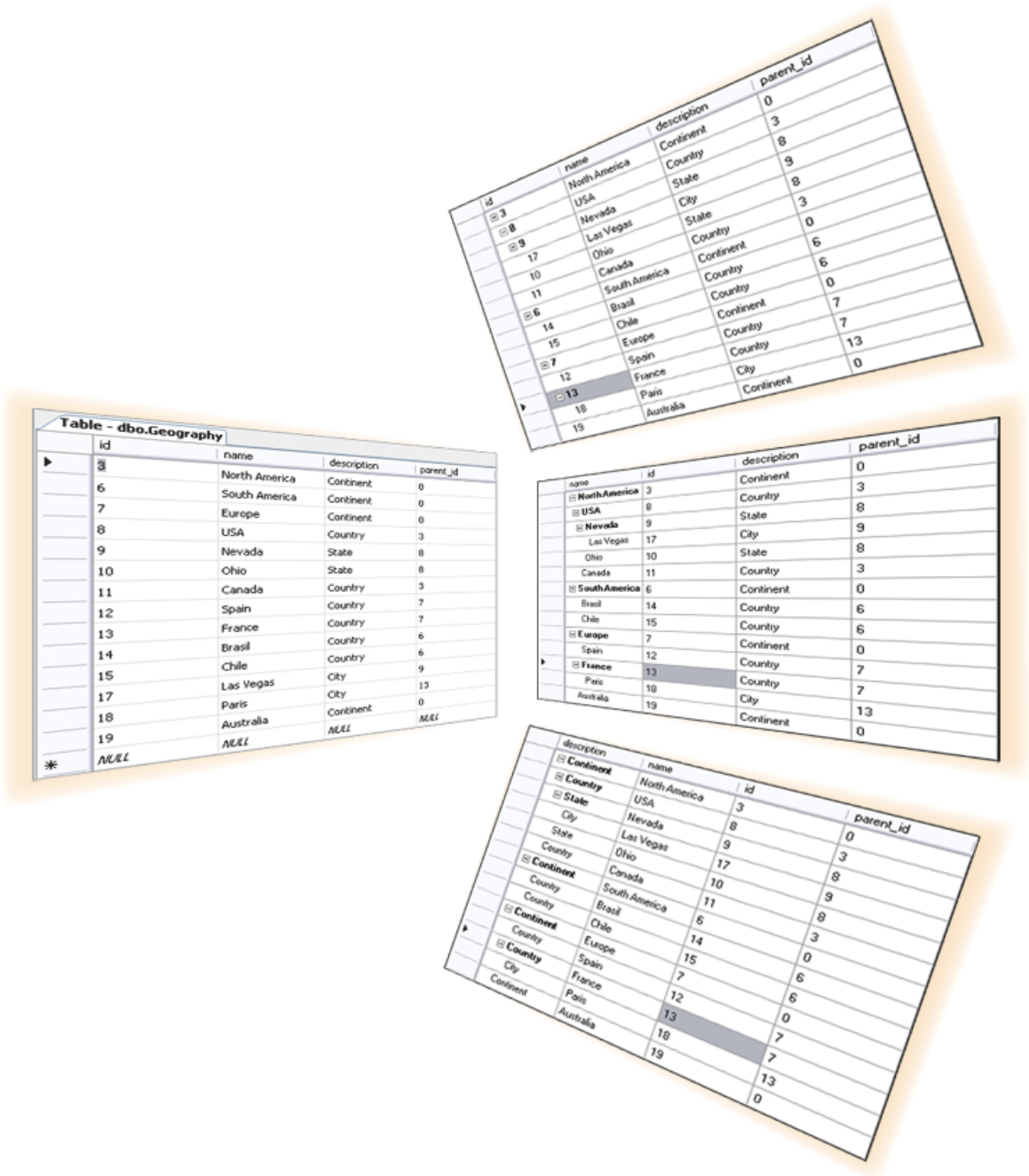
1. Creating internal table (System.Data.DataTable) that contains a copy of data set with DataSource property.
2. Expanding internal table with additional columns in which data needed for hierarchy viewing is being generated and saved.
3. Processing data set with DataSource property, using information set with PrimaryKeyColumnName, HierarchicalColumnName and RootValue, and filling additional columns of internal table with specific values. Data that is set by user is not being altered in any way during this process.
4. Adjusting DataSource property to DataGridView control so that it points to a recreated and filled internal table.
5. Drawing control based on information stored in internal table and used for rendering the elements of hierarchy.

For whom is intended Hierarchical DataGridView control?

The control is intended for all those users who need:

☑	Adaptable and efficient view of smaller amount of hierarchically organized data
☑	View of hierarchy without additional time spent on developing and filling in respective data structures
☑	View of data from various data sources (MS SQL Server, MS Access, XML and all the other sources to which the user can connect through ADO.NET)
☑	Simple and flexible control that does exactly what its name says: displaying hierarchically organized data

Figure 1 How does Hierarchical DataGridView work?



FEATURES

A special feature of Hierarchical DataGridView is its ability to analyze, process and display hierarchical data created in a flat structure (typically in System.Data.DataTable). At the same time developer does not have to make an effort to prepare and transform the data that he, for example, has got from relational database, into a form suitable for hierarchical viewing (e.g., he does not have to build and fill any collection of nodes, that reflects data hierarchy, to enable the viewing of data in this control).

General Features

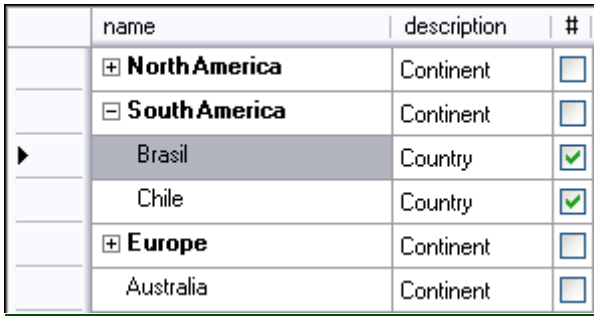
☑	Customizable table for displaying hierarchical data
☑	Not a single line of program code needs to be written in order to display hierarchy
☑	The style of data displaying and the look of Hierarchical DataGridView control are entirely adjustable
☑	Simple usage – all required features can be set up in graphic environment
☑	Shows data from any source by using ADO.NET and DataTable objects
☑	Bound, Unbound and Virtual mode
☑	Customizable style appearance and behavior
☑	Full customization of columns, rows, cells and borders

Special Features:

☑	This component supports direct data editing, arbitrary sorting and rearranging of columns without losing the hierarchical view
☑	Before / After events allow unlimited flexibility when using hierarchy
☑	The possibility of automatic data sorting without losing the hierarchical view
☑	Advanced sorting possibility with the ability to use custom sort indicators in column headers
☑	Conformed with Microsoft.NET Framework Design Guidelines
☑	Analyzed with Microsoft FxCop tool - Hierarchical DataGridView component has fulfilled all requirements

SCREENSHOTS

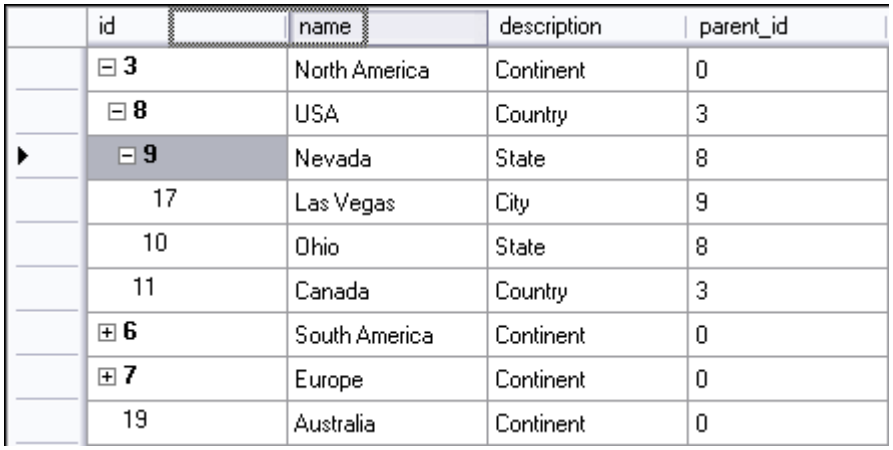
Figure 2 CheckBox



	name	description	#
	⊞ **North America**	Continent	☐
	⊟ **South America**	Continent	☐
▶	Brasil	Country	☑
	Chile	Country	☑
	⊞ **Europe**	Continent	☐
	Australia	Continent	☐

One of the Hierarchical DataGridView possibilities is to view hierarchical data with the full control of specifying how the data inside every single cell will be displayed. The control allows the usage of CheckBoxes, ComboBoxes, links, Buttons, calendars ...

Figure 3 ColumnReordering



	id	name	description	parent_id
	⊟ **3**	North America	Continent	0
	⊟ **8**	USA	Country	3
▶	⊟ **9**	Nevada	State	8
	17	Las Vegas	City	9
	10	Ohio	State	8
	11	Canada	Country	3
	⊞ **6**	South America	Continent	0
	⊞ **7**	Europe	Continent	0
	19	Australia	Continent	0

Regardless of whether the hierarchical data or flat data is being displayed in the control, you have full control over defining the order of columns - by simple drag and drop you can allow your users to specify, in a run-time, the order of columns on their own. If the hierarchical data is displayed in the control, then control will automatically take care of displaying the plus and minus signs.

Figure 4 DataEditing

	name	description
	Australia	Continent
	⊞ **Europe**	Continent
	⊟ **North America**	Continent
	Canada	Country
▶	USA	Country
	⊟ **Nevada**	State
	Las Vegas	City
	Ohio	State
	⊞ **South America**	Continent
*		

One of the features of this control is in-cell data editing. The control allows data editing even in the column which contains plus and minus signs. After adding a single line or deleting it, the control can be automatically refreshed in order to display newly entered data in the respective place within the hierarchy. The operations of changing, adding and deleting are allowed even in the case when the control does not display hierarchical data.

Figure 5 GUI Customization

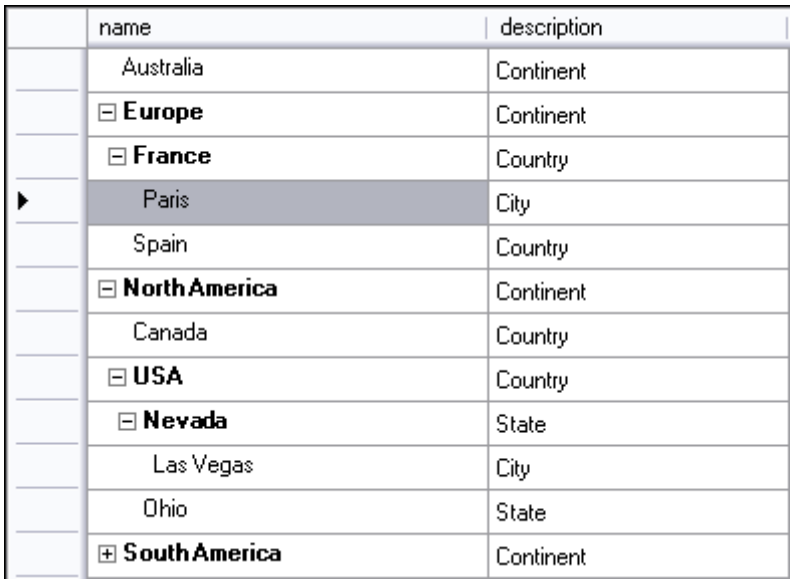
By setting up the certain properties, the user has the full control over the way how data will be displayed inside Hierarchical DataGridView control. In that process a special attention is paid to the adjustment of displaying column that contains plus and minus signs needed for browsing through the hierarchy.

Figure 6 HierarchyNot

id	name	description	id_parent
10	Brasil	Country	9
2	Canada	Country	1
11	Chile	Country	9
6	Europe	Continent	0
8	France	Country	6
4	Nevada	State	3
1	North America	Continent	0
5	Ohio	State	3
9	South America	Continent	0
7	Spain	Country	6
3	USA	Country	1

Hierarchical DataGridView can be used even in the situations when there is no need to display hierarchy. By simple setting up of the certain property, data will not be displayed in hierarchical form, but in normal, table-shaped (flat) form. In this case the control is functioning almost identically as the Microsoft DataGridView control.

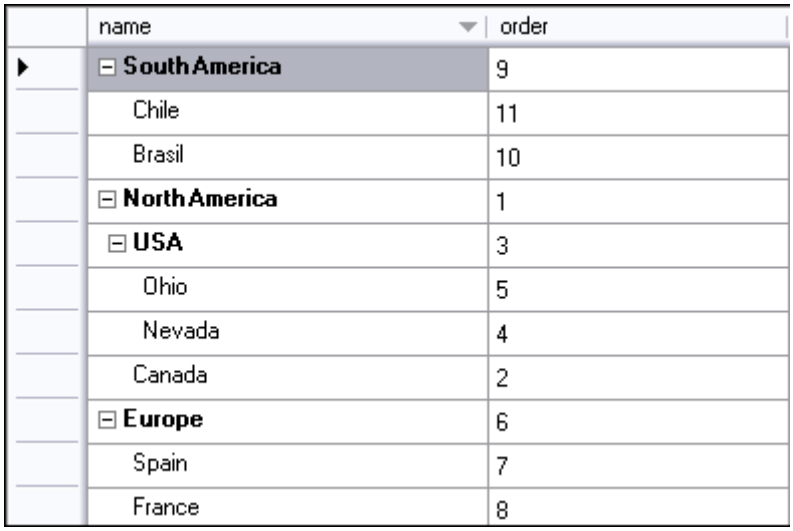
Figure 7 SimpleHierarchy



name	description
Australia	Continent
Europe	Continent
France	Country
Paris	City
Spain	Country
NorthAmerica	Continent
Canada	Country
USA	Country
Nevada	State
Las Vegas	City
Ohio	State
SouthAmerica	Continent

Hierarchical DataGridView control is created for displaying the relatively small amount of hierarchically organized data. With this control user does not need to write a single line of code to display data in a hierarchical form – all that he needs to do is to change the respective property.

Figure 8 Sort



name	order
SouthAmerica	9
Chile	11
Brasil	10
NorthAmerica	1
USA	3
Ohio	5
Nevada	4
Canada	2
Europe	6
Spain	7
France	8

The user has the possibility to adjust sorting of data displayed inside this control. At this point it is important to emphasize that the sorting, if made according to the column in which the hierarchy is displayed, considers the level at which the single data is located. In that way sorting is functioning properly even in those situations in which the classic sorting does not give expected results.

Figure 9 Unbound Column

id	name	description	parent_id	Sum of id and parent_id
⊟ **1**	North America	Continent	0	1
⊟ **4**	USA	Country	1	5
⊟ **5**	Nevada	State	4	9
12	Las Vegas	City	5	17
6	Ohio	State	4	10
7	Canada	Country	1	8
⊟ **2**	South America	Continent	0	2
10	Brasil	Country	2	12
11	Chile	Country	2	13
⊟ **3**	Europe	Continent	0	3
8	Spain	Country	3	11
⊟ **9**	France	Country	3	12
13	Paris	City	9	22
14	Australia	Continent	0	14

Despite the fact that displayed hierarchical data inside Hierarchical DataGridView control is taken from some other database, the user has a full freedom of adding the additional columns in which displayed data is not originally located in the data source. This means that the control allows adding of columns for displaying the most various data that is entirely independent on the main data source.

OTHER SCREENSHOTS

Figure 10 File Browser

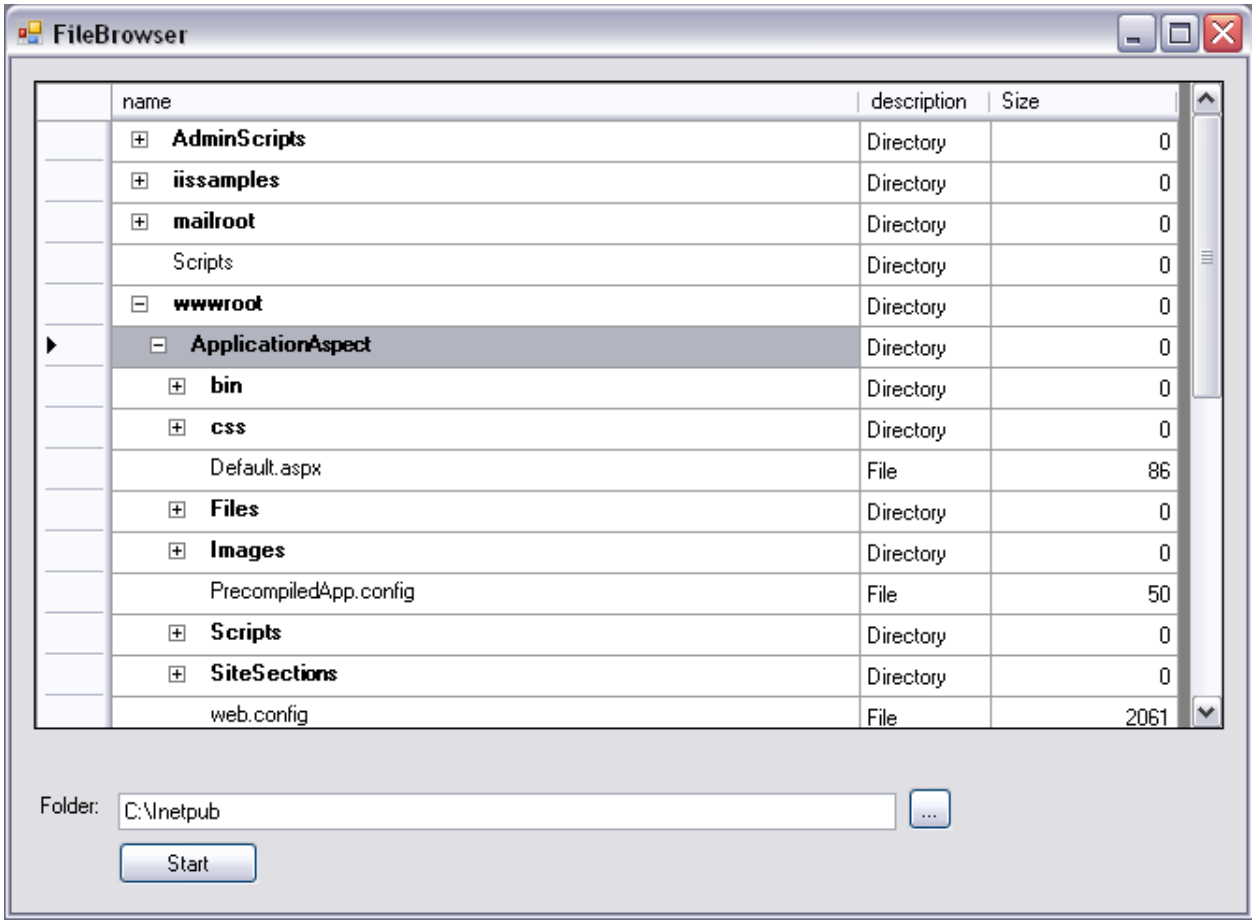


Figure 11 Master-Details

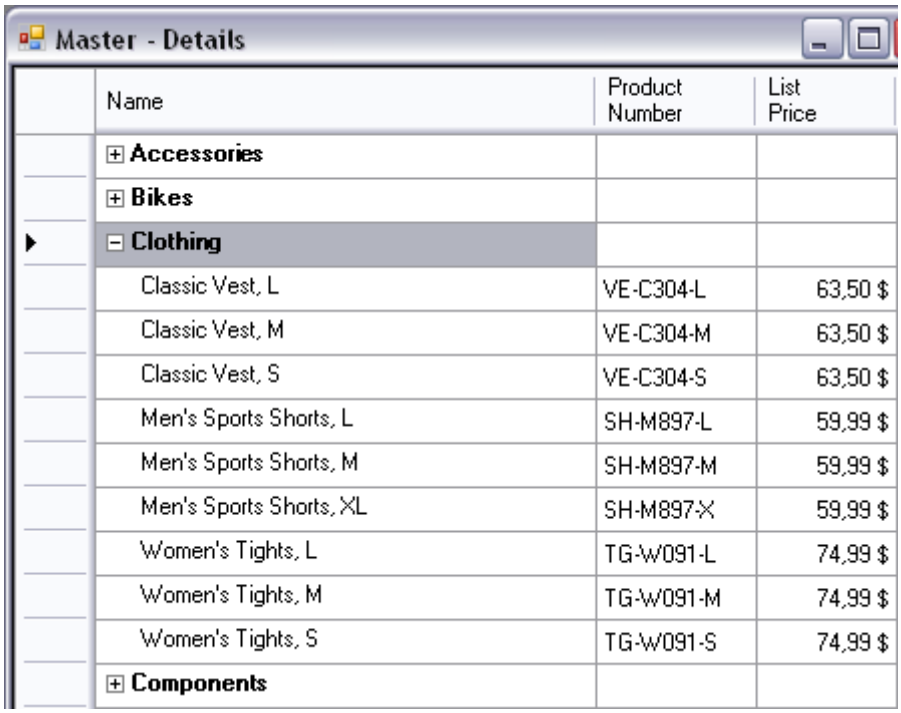


Figure 12 Microsoft Project

MS Project

Task name	Duration	Start	Finish
Five to Eight Weeks Before Moving	16 Days	Thu 1.1.04	Thu 22.1.04
Planning the Move	3 days	Thu 1.1.04	Mon 5.1.04
Calculate moving expenses	1 day	Thu 1.1.04	Thu 1.1.04
Determine the best method of moving	2 days	Fri 2.1.04	Mon 5.1.04
Create a moving-expense receipt file	1 day	Fri 2.1.04	Fri 2.1.04
Create a moving binder	1 day	Fri 2.1.04	Fri 2.1.04
Household Administration	6 days	Mon 5.1.04	Mon 12.1.04
Finances and Insurance	2 days	Mon 5.1.04	Tue 6.1.04
Review household finances	1 day	Tue 6.1.04	Tue 6.1.04
Verify that your belongings are insured for...	1 day	Mon 5.1.04	Mon 5.1.04
Appraise valuables specifically insured for...	2 days	Mon 5.1.04	Tue 6.1.04
Transfer insurance policies to your new add...	1 day	Mon 5.1.04	Mon 5.1.04
Transfer medical insurance to your new loca...	1 day	Mon 5.1.04	Mon 5.1.04
Notifications	2 days	Tue 6.1.04	Wed 7.1.04
Notify your employer of your moving dates a...	1 day	Tue 6.1.04	Tue 6.1.04
Transfer, sell, or discontinue any membersh...	2 days	Tue 6.1.04	Wed 7.1.04
Vital Services and Records	2 days	Thu 8.1.04	Fri 9.1.04
Organize important records	2 days	Thu 8.1.04	Fri 9.1.04
Contact your doctor, dentist, and vet for r...	1 day	Thu 8.1.04	Thu 8.1.04
Request additional refills of vital medicat...	1 day	Thu 8.1.04	Thu 8.1.04
Your New Residence	2 days	Fri 9.1.04	Mon 12.1.04
Collect shopping and dining guides from you...	2 days	Fri 9.1.04	Mon 12.1.04
Determine the location of the police, fire ...	1 day	Fri 9.1.04	Fri 9.1.04
Change drivers/vehicle license address if r...	1 day	Fri 9.1.04	Fri 9.1.04
Place orders for new furniture	2 days	Fri 9.1.04	Mon 12.1.04

Figure 13 Organizational Structure

Organizational Structure

Name	No. of Employees	No. of Servers	No. of Workstations
Company	1200	40	800
Production	700	20	400
Purchasing	100	5	100
Transportation	200	5	100
Railroad Administ...	150	3	80
Administrative Se...	50	2	20
Sales and Service	200	10	200

First try it, than buy it!" is Application Aspect philosophy. It is very important to us that our customers get what they have paid for.

Evaluate any of our products free of charge for unlimited trial period. There is no expiration date for the trial period. However, you will not be able to use the component in a production environment as it will generate a reminder, from time to time. Each evaluation is fully functional and comes with complete sample projects that include full source code, help documentation and tutorials.

Download it, evaluate it and compare it to what you are currently using, planning to use or what you are expecting.

PRICING

PRODUCT	PRICE
Hierarchical DataGridView - 1 Developer License	130,00 USD
Hierarchical DataGridView - 4 Developers License Pack	390,00 USD
Hierarchical DataGridView - 8 Developers License Pack	780,00 USD
Hierarchical DataGridView - Enterprise License (Allows Unlimited Developers at a Single Physical Address)	1300,00 USD

What am I buying?

You are buying license file that will allow you to use our product acording to the rights written in EULA comprising one year subscription that includes:

- E-mail support for the full duration of your subscription term
- Free minor/major version upgrades
- Free hotfixes and service packs

License Pack Benefits

- Pay 3 get 4 (1 free)
- Pay 6 get 8 (2 free)

Choosing license packs can be cheaper… Buying 4 or 8 developers license pack is better value than buying 4 or 8 single developer licenses. For example, if you are buying 4 licenses, buy license pack and get 1 license free or in case you are buying 3 licenses get one additional license to grow into.