



INFRAGISTICS

DESIGN / DEVELOP / EXPERIENCE

All samples, demos, slides & recording will be available later today for download at blogs.infragistics.com

I will tweet when they are posted, follow @jasonberes to be the first to get them.

Presenting the igHierarchicalGrid Control NetAdvantage for JQUERY

JASON BERES : VP Product Management
jasonb@infragistics.com, @jasonberes

TOM PUGLISI : Technical Evangelist
tomp@infragistics.com. : @tompuglisi

All samples, demos, slides & recording will be available later today for download at blogs.infragistics.com

I will tweet when they are posted, follow @jasonberes to be the first to get them.

Agenda

- Infragistics NetAdvantage for JQUERY igHierarchicalGrid introduction
- Cover igHierarchicalGrid features
- Demonstrate igHierarchicalGrid samples
- igHierarchicalGrid help, resources, and videos
- Q & A and Wrap-up

All samples, demos, slides & recording will be available later today for download at blogs.infragistics.com

I will tweet when they are posted, follow @jasonberes to be the first to get them.

igHierarchicalGrid

Multi-level, high-performance, feature-rich grid control

All samples, demos, slides & recording will be available later today for download at blogs.infragistics.com

I will tweet when they are posted, follow @jasonberes to be the first to get them.

Core Features

- Based on jQuery UI
- Theme Roller Support
- Lightweight, High Performance
- Paging, Sorting, Filtering, Editing
- Rich Client-Side Event APIs

Show 5 records

ID	Product Name	SKU #	Color																						
1	Adjustable Race	AR-5381																							
<table border="1"><thead><tr><th>ID</th><th>Location</th><th>Shelf</th><th>Bin</th><th>Quantity</th></tr></thead><tbody><tr><td>1</td><td>1</td><td>A</td><td>1</td><td>408</td></tr><tr><td>2</td><td>1</td><td>6</td><td>B</td><td>5</td><td>324</td></tr><tr><td>3</td><td>1</td><td>50</td><td>A</td><td>5</td><td>353</td></tr></tbody></table>				ID	Location	Shelf	Bin	Quantity	1	1	A	1	408	2	1	6	B	5	324	3	1	50	A	5	353
ID	Location	Shelf	Bin	Quantity																					
1	1	A	1	408																					
2	1	6	B	5	324																				
3	1	50	A	5	353																				
2	Bearing Ball	BA-8327																							
3	BB Ball Bearing	BE-2349																							
4	Headset Ball Bearings	BE-2908																							
5	Blade	BL-2036																							

1 - 5 of 65 records

prev Pg 1 of 13 next

Category ID	Category Name	Description
1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
3	Confections	Desserts, candies, and sweet breads
4	Dairy Products	Cheeses
5	Grains/Cereals	Breads, crackers, pasta, and cereal
6	Meat/Poultry	Prepared meats
7	Produce	Dried fruit and bean curd

Product ID	Product Name	Quantity Per Unit
7	Uncle Bob's Organic Dried Pears	12 - 1 lb pkgs.
14	Tofu	40 - 100 g pkgs.
28	Rössle Sauerkraut	25 - 825 g cans
51	Manjimup Dried Apples	50 - 300 g pkgs.
74	Longlife Tofu	5 kg pkg.

Additional Features

- Load on Demand
- Virtualization
- Outlook™ GroupBy
- Summaries
- Filtering
- Column Chooser
- In-Place Editing
- Row State Tracking
- Interactive UI Elements

	Product ID	Name	Product Number	Color
	▽ Equals...	▽ Contains...	▽ Contains...	▽ Contains...
	+ Add new row			
+ 1	1	Adjustable Race	AR-5381	
+ 2	2	Bearing Ball sphere	BA-8327	
+ 3	3	BB Ball Bearing	BE-2349	
+ 4	4	Headset Ball Bearings	BE-2908	
	Count = 4.00	Count = 4.00	Count = 4.00	Count = 4.00
	Min = 1.00			
	Max = 4.00			
	Sum = 10.00			
	AVG = 2.50			

	Product ID	Name	Product Number
	▽ Equals...	▽ Contains...	▽ Contains...
	+ Add new row		
+ 1	1	Adjustable Race	AR-5381
+ 2	2	Gear	BA-8327
+ 3	3	BB Ball Bearing	BE-2349
+ 4	4	Headset Ball Bearings	BE-2908

Drag a column here or [select columns](#) to Group By

Column Chooser

- Hide Product ID
- Hide Name
- Hide Product Number
- Hide Color

Apply Cancel

API Fun

Enabling & Configuring Features:

```
$("#grid1").igHierarchicalGrid({
    features: [
        {
            name: "Paging",
            type: "local",
            pageSize: 4
        },
        {
            name: "Sorting"
        },
        {
            name: "Filtering"
        },
        {
            name: "Resizing",
            allowDoubleClickToResize:true
        }...
    ]
});
```

API Fun

Configuring Grid Levels:

```
autoGenerateColumns: false,  
  primaryKey: "EmployeeID",  
  columns: [  
    { key: "EmployeeID", headerText: "${EmployeeID}", dataType: "number", width: "100px" },  
    { key: "LastName", headerText: "${lbl_last_name}", dataType: "string", width: "150px" },  
    { key: "FirstName", headerText: "${lbl_first_name}", dataType: "string", width: "150px" },  
    { key: "HomePhone", headerText: "${HomePhone}", dataType: "string", width: "150px" }  
  ],  
  childrenDataProperty: "Orders",  
  autoGenerateLayouts: false,  
  columnLayouts: [  
    {  
      key: "Orders",  
      responseDataKey: 'results',  
      autoGenerateColumns: false,  
      primaryKey: "OrderID",  
      columns: [  
        { key: "OrderID", headerText: "${OrderID}", dataType: "number", width: "50px" },  
        { key: "CustomerID", headerText: "${CustomerID}", dataType: "string", width: "60px" },  
        { key: "Freight", headerText: "${Freight}", dataType: "string", width: "70px" },  
        { key: "ShipName", headerText: "${ShipName}", dataType: "string", width: "100px" },  
        { key: "ShipAddress", headerText: "${ShipAddress}", dataType: "string", width: "90px" },  
        { key: "ShipCity", headerText: "${ShipCity}", dataType: "string", width: "90px" },  
        { key: "ShipCountry", headerText: "${ShipCountry}", dataType: "string", width: "80px" }  
      ].....  
    }  
  ]
```


What Do Our Customers Think?





Fascinating. How do I get Data into this thing?

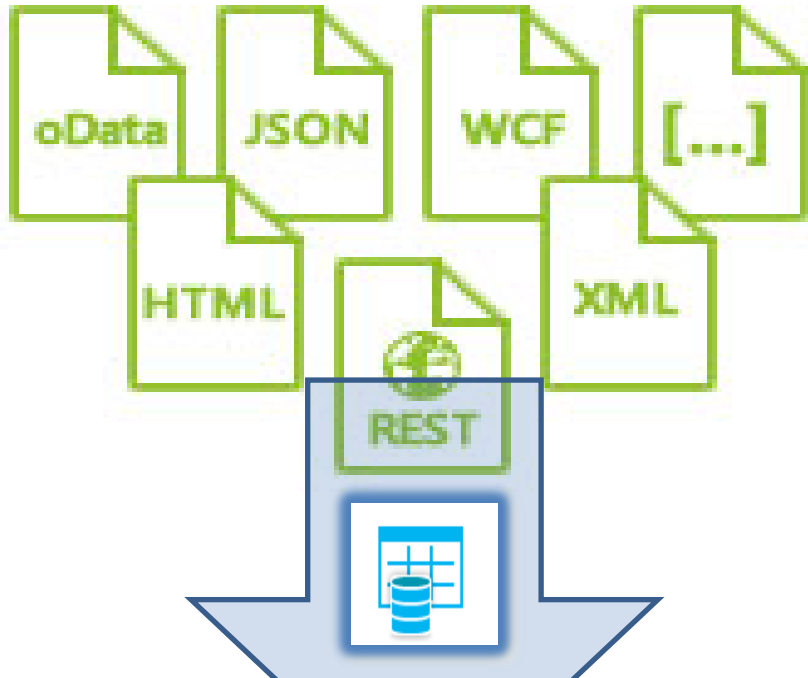
igDataSource

The bridge between many data source types and your jQuery controls



igDataSource

- Access various data sources
- Provides consistent data interface to grid
- Provides paging, sorting, filtering capabilities
- Can mash-up disparate data sources into one schema



Category ID	Category Name	Description
+ 1	Beverages	Soft drinks, coffees, teas, beers, and ales
+ 2	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
+ 3	Confections	Desserts, candies, and sweet breads
+ 4	Dairy Products	Cheeses
+ 5	Grains/Cereals	Breads, crackers, pasta, and cereal
+ 6	Meat/Poultry	Prepared meats
- 7	Produce	Dried fruit and bean curd
Product ID Product Name Quantity Per Unit		
7	Uncle Bob's Organic Dried Pears	12 - 1 lb pkgs.
14	Tofu	40 - 100 g pkgs.
28	Rössle Sauerkraut	25 - 825 g cans
51	Manjimup Dried Apples	50 - 300 g pkgs.
74	Longlife Tofu	5 kg pkg.
+ 8	Seafood	Seaweed and fish



igDataSource

The igDataSource is a jQuery-based component which serves as an intermediate layer between the data bound controls - including the igGrid and other Infragistics controls - and the actual data source.

Data Formats

Server

- Data (Web Services)
- REST GET
- WCF REST (JSON & XML)
- JSONP Service
- Any REST Service

Client

- JSON
- XML
- JavaScript Object Arrays
- Strings of XML or JSON
- JavaScript functions returning data in any of the formats above
- Existing HTML structures (such as a HTML table filled with data)
- Mashup scenarios

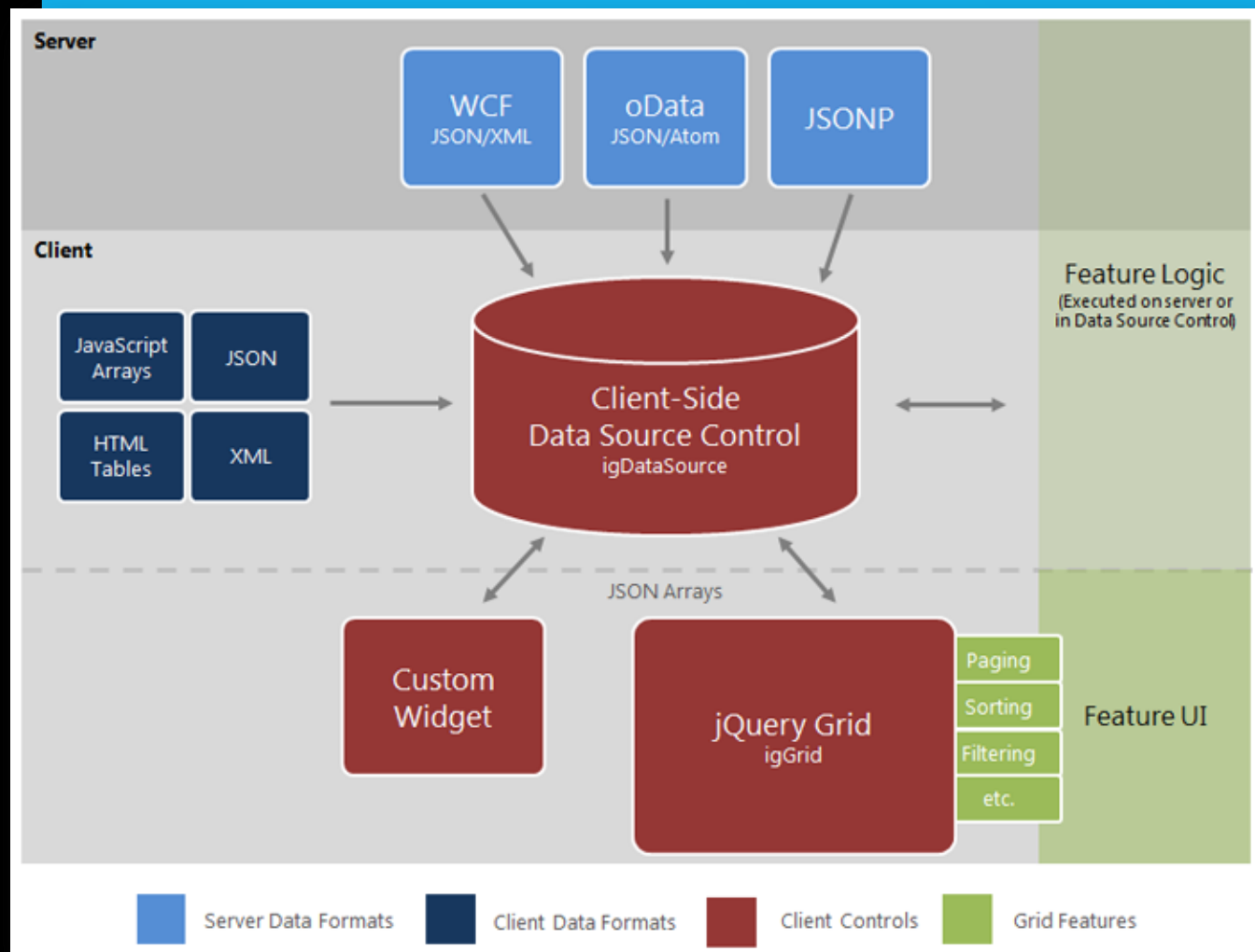
Supported Features:

- Local and remote data Paging
- Sorting
- Filtering
- Data schema support: translate data from many sources into a consistent format



Architecture

Extensions can be created for custom data sources, but most common scenarios are covered in the control



API Fun

Configuring igDataSource for oData:

```
var ds,
    oDataSchema,
    render;

render = function (success, error) {
    if (success) {
        $("#t1 tbody").empty();
        $("#tableTemplate").tmpl(ds.dataView()).appendTo("#t1 tbody");
    } else { alert(error); }
}

$.ig.loader({
    scriptPath: "<% = Url.Content("~/samplescommon/jquery/common/js") %>",
    cssPath: "<% = Url.Content("~/samplescommon/jquery/common/css") %>",
    resources: "igShared,igDataSource"
});

$.ig.loader(function () {
    oDataSchema = new $.ig.DataSource("json", { fields: [
        { name: "ProductID", type: "number" },
        { name: "Name", type: "string" },
        { name: "ProductNumber", type: "string" },
        { name: "StandardCost", type: "number"}],
        searchField: "d"
    });

    var url= /samplesbrowser/Services/AdventureWorksDataService.svc/Products?$format=json&$top=10";

    ds = new $.ig.DataSource({ type: "remoteUrl", callback: render, dataSource:
    url, schema: oDataSchema,
    responseDataKey: "d", responseType: "jsonp" });

    ds.dataBind();
```

API Fun

Assign igDataSource to Grid

```
$("#grid1").igGrid({
```

```
....
```

```
    dataSource: ds
```

```
....
```


Demo Time!

Let's check out some samples and some online references / resources!

- [Samples](#)
- Docs
 - [Walkthrough Topics](#)
 - [API](#)
- [Videos](#)

igHierarchicalGrid Summary

- Fast, high-performance, lightweight
- Built on JQuery UI
- Feature rich
- More features coming
- Part of our ever-expanding NetAdvantage for JQUERY product
- Used in MANY types of apps
- Used in
 - ASP.NET (client-side)
 - ASP.NET MVC (we give you the MVC Wrappers!)
 - PHP, ASP Classic, or just ANY web platform!

Resources



Download the product today!

- <http://infragistics.com/jquery>
- <http://www.infragistics.com>
- <http://community.infragistics.com>
- <http://samples.infragistics.com>
- Follow Me Twitter for Demos, HTML5 & jQuery Tips

@jasonberes

- Additional questions – jasonb@infragistics.com



INFRAGISTICS
DESIGN / DEVELOP / EXPERIENCE

QA & Wrap-Up



INFRAGISTICS
DESIGN / DEVELOP / EXPERIENCE

Thank You !!