# gravitybox.com

## Top-notch Scheduling Components
### Designed for the Medical Industry

Gravitybox Schedule.NET is a fully functional scheduling component designed for the medical industry. It allows a user to easily add, edit and delete appointments from an intuitive interface. The display can be configured to display appointment data in many different ways. Behind the appealing facade is a robust object model allowing developers great power over object manipulation and display.

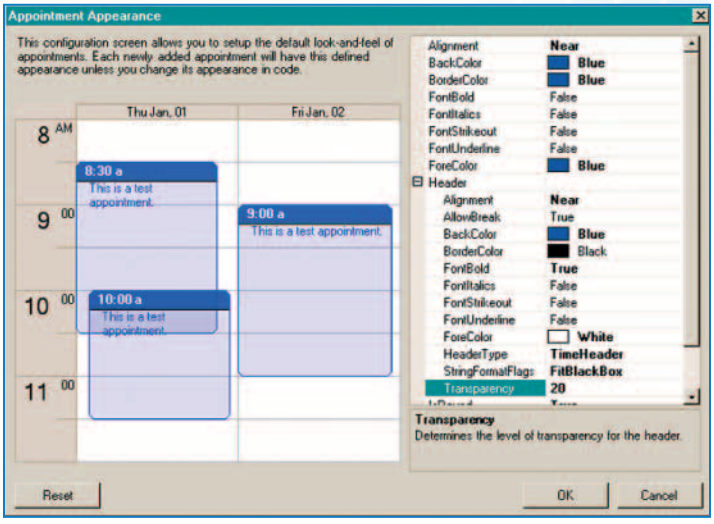
## APPEARANCE

Appointments can be made to appear in almost any fashion conceived. An Appearance object controls every aspect of each appointment, including back color, fore color, border color, alignment, font, etc. A single appearance object can be associated to multiple appointments, thus creating appearance groups. Change one property on the appearance object and all associated appointments are redrawn with the new setting. Appointments can be set to appear transparent, consequently allowing objects underneath them to be seen.

Each appointment can show a header as well. This is useful for providing summary information to the user in a fashion that is consistent with all or a group of appointments. Each appointment header has its own Appearance object to control its display. Again a single appearance object may be assigned to multiple headers creating appearance groups if desired.

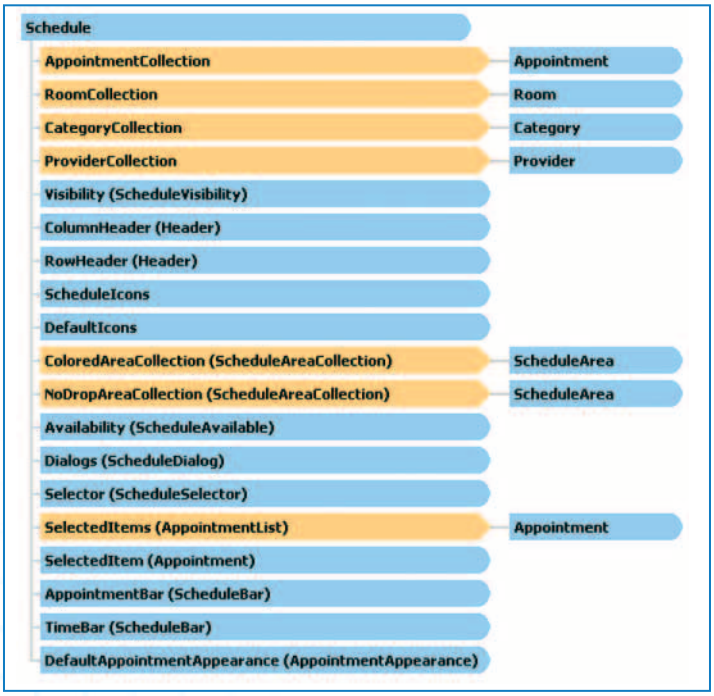
Furthermore, rows and columns have their own header definitions to identify their display. They may be formatted with their own font, colors, and transparency. This allows for complete control over the look-and-feel of a schedule.

The user may select a single appointment or any number of appointments. Selected appointments appear with thicker borders. You can perform actions on a group of appointments defined by the user through the SelectedItems collection, which holds a reference to each appointment the user has highlighted.



# OBJECT MODEL

The schedule object model is simple enough to be understood easily but robust enough to handle almost any situation. The key ingredient is flexibility. There are four main collections: Appointment, Room, Category, and Provider. The AppointmentCollection provides an interface to maintain a set of appointments. Rooms are held in the RoomCollection and can be used to group appointments for the same date. Most physicians' offices schedule appointments by date and room. This collection provides a built-in way to display and maintain rooms. The CategoryCollection allows you to group appointments by some arbitrary set of categories. Appointments can be clustered into groups like Surgery, Cleaning, Misc, etc. The ProviderCollection offers an integrated approach to handling employee assignment to appointments.

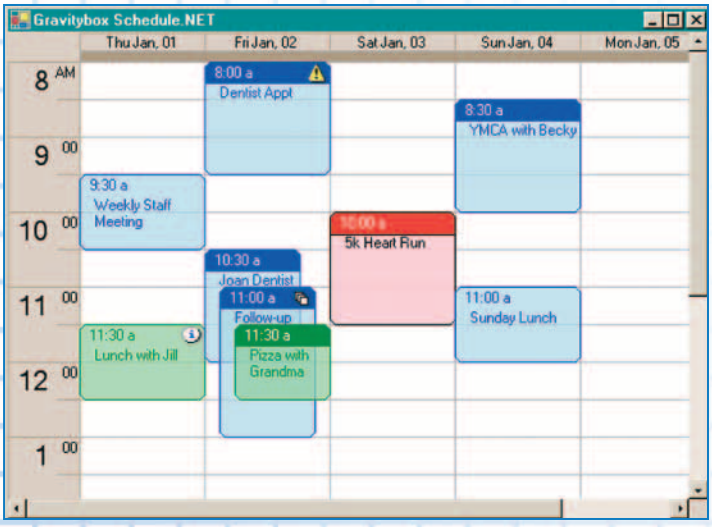


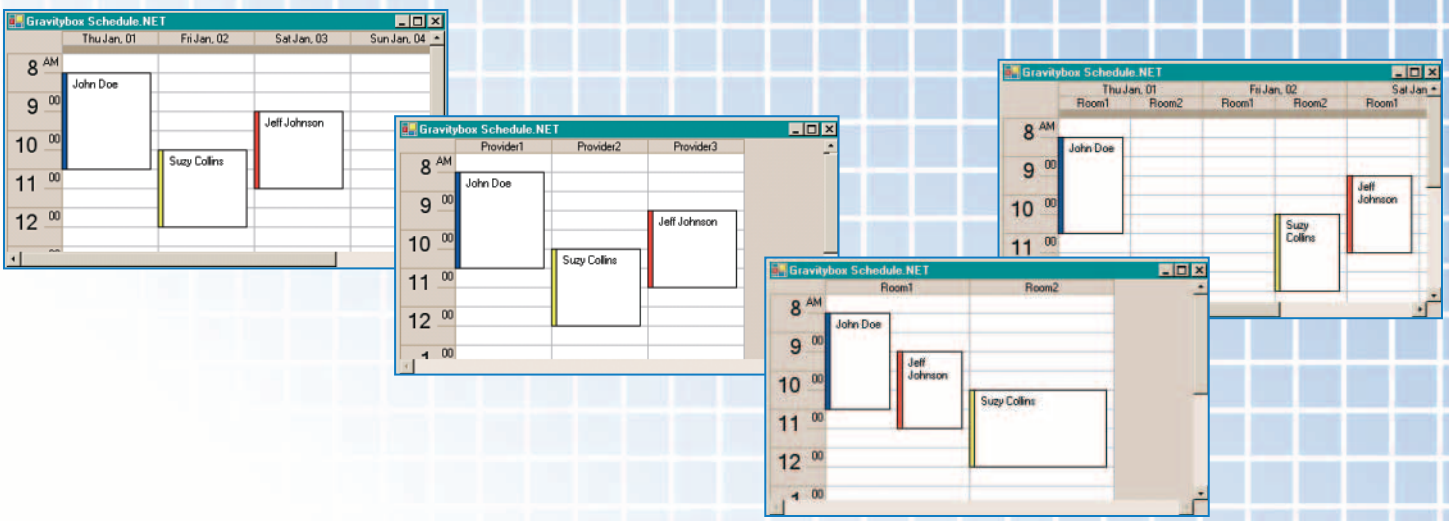

# VIEWMODES

The single-most impressive functionality of Gravitybox Schedule.NET is its ability to display the same information in multiple ways. The viewmode property allows appointments to be displayed based on a multitude of criteria. The most common data view is dates on top and times on left. This is a standard appointment book view. However in a physician's office, appointments are most likely scheduled into rooms as well. The schedule can display date and room on top and time on left. There are a total of 15 viewmodes that display dates, rooms, times, and providers in various combinations.

# CONFLICTS

Conflict resolution has been fully integrated into the schedule from its inception. This is one of the most difficult situations to display on a schedule properly. Gravitybox Schedule.NET provides multiple, conflict behaviors. Appointments may be placed overlapped, side-by-side, or staggered. The most common display mode is side-by-side. It allows the whole of every appointment to be viewed with no overlapping sections under any circumstances. A newer, visually interesting display is staggered. This setting tiles conflicting appointments in such a way as to ensure that at least some portion of each appointment is viewable but not all of an appointment. This is most attractive when used with a partially transparent background.






## VISIBILITY & AVAILABILITY

The display of appointments is not very useful without a powerful backend. Gravitybox Schedule.NET provides helper objects to query and search the schedule quite easily. For example, the Visibility object allows a developer to easily find the first visible row, column, date, room, time, or provider. Any one of these values may be retrieved from a set of coordinates as well. The reverse is also true, given a row, column, date, room, time, or provider coordinates the underlying object can be identified. Methods also exist to determine if a specific row, column, etc is visible in the viewport. Lastly, you may guarantee that a row, column, etc is visible by calling a method like ShowDate, ShowTime, and the like.

The Availability object allows you to query the schedule for the next available slot. This is a godsend for any software developer who has tried to write an algorithm to compute used and available time-slots on a schedule. Multi-day appointments, events, and conflicts compound the difficulty of writing an efficient algorithm to perform this action. The IsAreaAvailable method allows you to easily determine if a specific area of the schedule is free or taken.

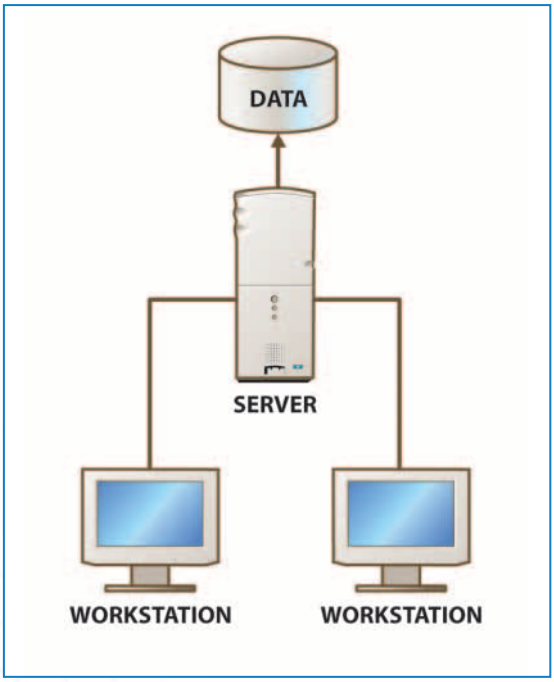
The NoDropAreaCollection defines areas of the schedule where appointments cannot be scheduled. This is useful to define holidays, weekend, or lunch-hours. You may even use it to mark a room as off-limits while it is being remodeled. Since there are restricted areas on a schedule, a way exists to check for restrictions. The IsAreaEnabled method of the Availability object allows you to determine if a schedule area is restricted or not.

## COMPLEX QUERIES

Finding an appointment for a client is simple. A database query can be written to find all appointments for John Doe. However many times it is necessary to find appointments based on more complex information. It would be cumbersome to write a database query to find all appointments between 9:00am and 12:00pm on April 23 with both Tom and Judy as the appointment providers, scheduled in room “OP-2” and assigned to the “Surgery” category. This is quite complex but completely possible with out-of-the-box functionality. The Find method of the AppointmentCollection returns an appointment list. The method is overloaded to take many parameter types. To create complex queries, find each of these criteria singly and intersect the returned lists. This intersection is the list for the defined criteria. In fact, all lists have three defined operations: Intersect, Subtract, and Union. These may be used to construct arbitrarily complex queries.

# NETWORKING

The future of Gravitybox Schedule.Net lies in networking. In development currently, is a series of web services that allow multiple schedule clients to coordinate data with each other or a centralized database repository. This capability allows enterprise functionality to be added to any application with minimal effort. A schedule client, through a web service, may access a server database. This allows coordination of data through an existing back-end allowing you to free development resources for other endeavors.




# CONCLUSION

Gravitybox Schedule.NET was designed from the ground up with a physician's office in mind. Significant effort was made to ensure that it is easy to use for a medium level software developer, such as properties that are intuitively named. There is no confusion as to what properties and methods do. The bottom line is this component contains tens of thousands of lines of code of debugged and ready to use now! Anyone who has attempted to write a scheduling system from the ground up, knows that is an error prone and time-consuming process. Having a plug-in component greatly decreases your application's time-to-market. Time-to-market is the life or death of any business.

gravitybox.com
1.866.GRAVBOX

Voice-Mail and Fax (USA):
(+1) 866.GRAVBOX
(+1) 866.472.8269

Gravitybox Software LLC
550 Arncliffe Court
Suite 100
Alpharetta, GA 30005 USA

www.gravitybox.com
www.gravitybox.co.uk

feedback@gravitybox.com