= Something ' " , or a compound condition such as " title =
'Title 1 ' | | title
= ' Title 2 ' & & author =
'Someone ' " . These boolean and relational operations can be specified using C # like & & , | | , = , ! = operators as well as their English @-@ like equivalent like EQUAL , NOT EQUAL . SQL like operators such as LIKE , GROUP BY and ORDER BY are also supported , as are wildcard conditions . So , " title LIKE ' any * ' " is a valid query string . These operators can be used to execute complex searches such as

The above code snippet creates an ItemSearcher object that searches on the OutContactRelationship instance that relates pictures and contacts , in effect searching all pictures related with a contact . It then runs the query Name LIKE ' A * ' " on all contacts reachable through OutContactRelationship , returning the list of " contacts whose names start with A and whose pictures I have " . Similarly , more relationships could be taken into account to further narrow down the results . Further , a natural language query processor , which parses query in natural language and creates a well @-@ formed OPath query string to search via proper relationships , can allow users to make searches such as " find the name of the wine I had with person X last month " , provided financial management applications are using WinFS to store bills .

Different relations specify a different set of data . So when a search is made which encompasses multiple relations , the different sets of data are retrieved individually and a union of the different sets is computed . The resulting set contains only those data items which correspond to all the relations .

= = = Notifications = = =

WinFS also includes better support for handling data that changes frequently . Using WinFS Notifications , applications choose to be notified of changes to selected data Items . WinFS will raise an ItemChangedEvent , using the .NET Event model , when a subscribed @-@ to Item changes , and the event will be published to the applications .

= = Data sharing = =

WinFS allows easy sharing of data between applications , and among multiple WinFS stores , which may reside on different computers , by copying to and from them . A WinFS item can also be copied to a non @-@ WinFS file system , but unless that data item is put back into the WinFS store , it will not support the advanced services provided by WinFS .

The WinFS API also provides some support for sharing with non @-@ WinFS applications . WinFS exposes a shell object to access WinFS stores . This object maps WinFS items to a virtual folder hierarchy , and can be accessed by any application . WinFS data can also be manually shared using network shares , by sharing the legacy shell object . Non @-@ WinFS file formats can be stored in WinFS stores , using the File Item , provided by WinFS . Importers can be written , to convert specific file formats to WinFS Item types .

In addition , WinFS provides services to automatically synchronize items in two or more WinFS stores , subject to some predefined condition , such as " share only photos " or " share photos which have an associated contact X " . The stores may be on different computers . Synchronization is done in a peer @-@ to @-@ peer fashion ; there is no central authority . A synchronization can be either manual or automatic or scheduled . During synchronization , WinFS finds the new and modified Items , and updates accordingly . If two or more changes conflict , WinFS can either resort to automatic resolution based on predefined rules , or defer the synchronization for manual resolution . WinFS also updates the schemas , if required .

= = Application support = =

= = = Shell namespace = = =

WinFS Beta 1 includes a shell namespace extension , which surfaces WinFS stores as top level objects in My Computer view . Files can be copied into and out of the stores , as well as applications can be directly used to save there . Even folders such as My Documents can be redirected to the stores . WinFS uses Importer plug @-@ ins to analyze the files as they were being imported to the store and create proper WinFS schemas and objects , and when taking the objects out , re @-@ pack them into files . If importers for certain files are not installed , they are stored as generic File types .

= = = Microsoft Rave = = =

Microsoft Rave is an application that shipped with WinFS Beta 1 . It allows synchronization of two or more WinFS stores , and supports synchronization in full mesh mode as well as the central hub topology . While synchronizing , Microsoft Rave will determine the changes made to each store since the last sync , and update accordingly . When applying the changes , it also detects if there is any conflict , i.e. , the same data has been changed on both stores since the last synchronization . It will either log the conflicting data for later resolution or have it resolved immediately . Microsoft Rave uses peer @-@ to @-@ peer technology to communicate and transfer data .

= = = StoreSpy = = =

With WinFS Beta 1 , Microsoft included an unsupported application called StoreSpy , which allowed one to browse WinFS stores by presenting a hierarchical view of WinFS Items . It automatically generated virtual folders based on access permissions , date and other metadata , and presented them in a hierarchical tree view , akin to what traditional folders are presented in . The application generated tabs for different Item types . StoreSpy allowed viewing Items , Relationships , MultiSet , Nested Elements , Extensions and other types in the store along with its full metadata . It also presented a search interface to perform manual searches , and save them as virtual folders . The application also presented a graphical view of WinFS Rules . However , it did not allow editing of Items or their properties , though it was slated for inclusion in a future release . But the WinFS project was cut back before it could materialize .

= = = Type Browser = = =

WinFS also includes another application , named WinFS Type Browser , which can be used to browse the WinFS types , as well as visualize the hierarchical relationship between WinFS types . A WinFS type , both built @-@ in types as well as custom schemas , can be visualized along with all the properties and methods that it supports . It also shows the types that it derives from as well as other types that extend the type schema . However , while it was included with WinFS , it was released as an unsupported tool .

= = = OPather = = =

WinFS Beta 1 also includes an unsupported application , named OPather . It presents a graphical interface for writing Opath queries . It can be used by selecting target object type and specifying the parameters of the query . It also includes Intellisense @-@ like parameter completion feature . It can then be used to perform visualization tasks like binding results of a query to a DataGrid control , create views of the data in WinFS itself , or just extract the query string .

= = = Project " Orange " = = =

Microsoft launched a project to build a data visualization application for WinFS . It was codenamed

" Project Orange " and was supposedly built using Windows Presentation Foundation . It was supposed to provide exploration of Items stored in WinFS stores , and data relationships were supposed to be a prominent part of the navigation model . It was supposed to let people allow organization of the WinFS stores graphically as well ? productizing many of the concepts shown in the IWish Concept Video WMV File . However , since the WinFS project went dark , the status of this project is unknown .