Saved Filters – M1

Product Group	DMX
Last Updated	16 Jun 2009
Document Status	In Progress
Document Version	1
Document Owner	Santosh Chapaneri
Document Location	http://WLX/devspecs/Forms/AllItems.aspx
Windows Live Wave	4
Milestone	M1
Priority	(2) Normal

Primary Developer	Santosh Chapaneri	sachapan	**Program Manager**	Stacia Scott	staciasc
			Tester	Shanjeef Satchithanantham	shsatchi
Dev Lead	Ignatius Setiadi	ignatius	**Other**	Name	Alias

I. Document History

Date	Author	Description
05/28/2009	*Santosh Chapaneri*	*Created.*

II. Review History

Date	Reviewers	Result
mm/dd/yyyy	*Group, alias, …*	

III. Related Documents

Document	Owner	Location
PM One Pager	*StaciaSc*	*http://windows/wlx/4/pmspecs/Saved%20Filters%20-%20W4%20Page%20One.pptx*

Contents

1. Problem Statement

Windows Live Photo Gallery currently has great filtering capabilities, but we need a mechanism to save these filters so that the users can quickly find the photos/videos they are looking for. This is the Design Spec for M1 Saved Filters backend work. The PM spec can be found at http://windows/wlx/4/pmspecs/Saved%20Filters%20-%20W4%20Page%20One.pptx

1.1 Goals, Requirements, and Constraints

See PM spec for goals.

2. Solution Overview

2.1 Overview

Users can apply different filters and save these filters for later use. Since the UI for the saved filters will be adopted M2 onwards, the M1 work will involve only the backend management of the saved filters. This will be a debug-only feature in M1.

A user-hidden XML file will be created to save the specified filters. XML helper functions will be used to read/write the XML file. In the current architecture of DataSetBuilder, we are using three filter objects: FilterText, FilterRating and NavTreeFilter. These will correspond to the three separate nodes in XML file with their corresponding attributes.

In M2, the XML will be tweaked as per the UI design.

3. Solution Details

3.1 Sample XML file

A user-hidden XML file will be used to save the filter. This XML file will be stored at the location – C:\Users\<user name>\AppData\Local\Microsoft\Windows Live Photo Gallery\Saved Filters\SavedFilterFileName.xml

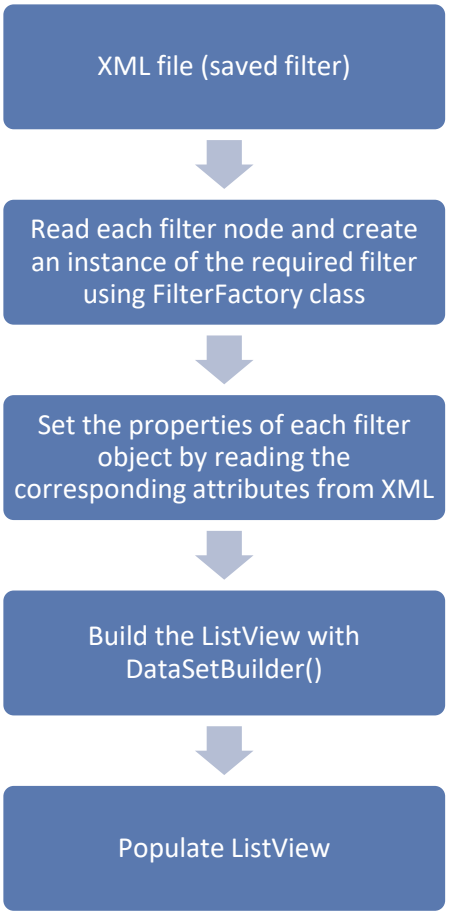
A sample XML file is shown below:

```
<?xml version="1.0" ?>
<FilterText name="Wordwheel" query="foo" />
<FilterRating name="Rating" minRating="2" maxRating="4" />
<FilterIntersection name="NavTree">
    <FilterUnion>
        <FilterFolder pathId="22" recursive="true" />
    </FilterUnion>
    <FilterUnion>
        <FilterLabel labelId="60" recursive="true" />
        <FilterLabel labelId="80" recursive="true" />
    </FilterUnion>
    <FilterUnion>
        <FilterDateRange dateBegin="mm/dd/yyyy" dateEnd="mm/dd/yyyy"
    </FilterUnion>
```

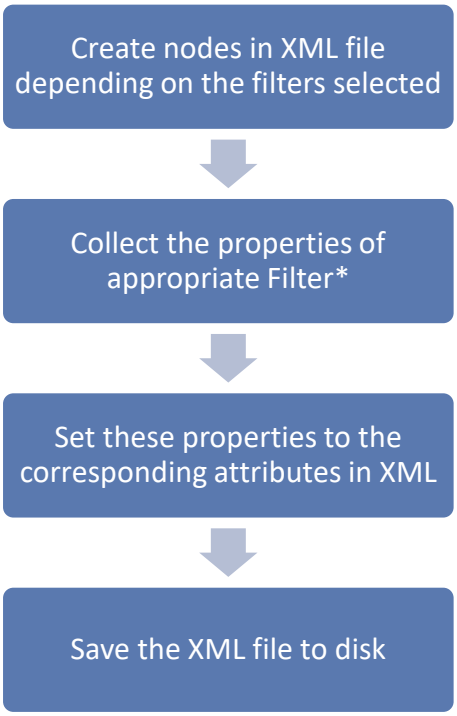
```
</FilterIntersection>
```

3.2 Supporting Diagrams

Parsing XML file to create the ListView



Saving the filter to an XML file



3.3 Modifications to existing Filter* classes

Each existing Filter* class will have an added support to:

a) Initialize itself from an XML element: instantiate the appropriate filter object and set its attributes from the corresponding XML node

b) Emit itself to an XML element: create a child node in XML and set its attributes with those of the filter

3.4 FilterFactory class

A FilterFactory class will be used to create the required instance with GetInstance() from the existing filter classes. The logic for initializing a FilterCollection from an XML element will give the child's element name to FilterFactory.

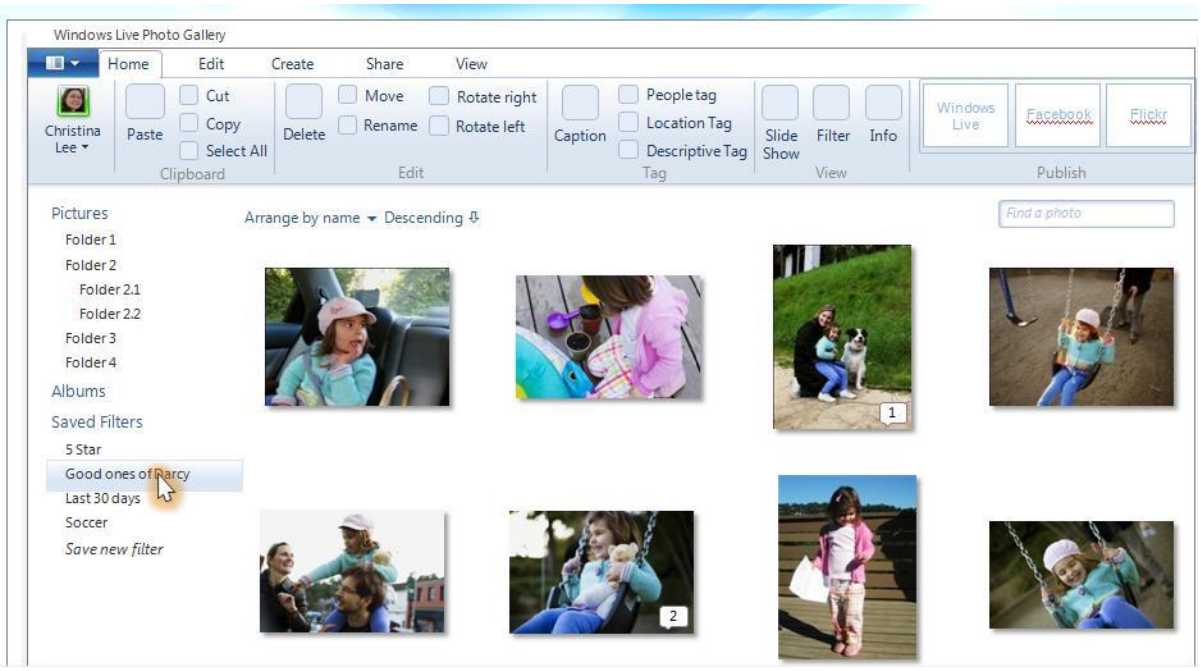
The filter will be instantiated only if the filter is selected by the user (saving) or if the filter node is present in XML (parsing).

3.5 FilterXmlManifestFactory class

FilterXmlManifestFactory class will be used to create an XML document, and add the required filter nodes. (The reusable logic and helper functions from PublishXmlManifest* can be used here by splitting out the common code to a shared base class, and the required functionality can be extended for FilterXmlManifestFactory.)

3.6 Saved Filters debug-only node

A Saved Filters (debug-only) node will be created in the left NavPane to show the list of saved filters and wil have an ability to save a new filter:



On clicking *Save new filter*, we will save the filters to an XML file at the location described above.

3.7 Assumptions, Dependencies, and Risks

3.7.1 Dependencies

Existing code for filtering will be used to populate the ListView.

3.7.2 Risks

3.7.3 Assumptions

3.8 Internal and External Special Values

3.9 Performance

3.10 Scalability

3.11 Privacy

3.12 Internationalization and Localization

3.13 Accessibility

3.14 Instrumentation & Monitoring

3.15 Testable Units

Order	Testable Unit
1	***XML parser***
2	***Integration of Saved Filters as a debug-only feature***

3.15.1 XML parser

3.15.1.1 Work Items

Description	Days	Pri	Dev
Modify existing Filter* classes	1.5	1	SaChapan
Add new FilterXmlManifestFactory class	0.5	1	
Add new FilterFactory class	0.5	1	
Serialization of filters to XML	1.5	1	
Parsing XML and rebuilding the filter tree	1.5	1	
Unit tests	0.75	1	

3.15.1.2 Dependencies

Existing filter source code

3.15.1.3 Exit Requirements

XML parsing functionality exercised and checked in.

3.15.2 Integration of Saved Filters as a debug-only feature

3.15.2.1 Work Items

Description	Days	Pri	Dev
Create debug-only Saved Filters node in NavPane	0.25	1	SaChapan
Hooking up XML parser to DataSetBuilder/ListView	1	1	
Display the list of saved filters and have an ability to save new filters	1	1	

3.15.2.2 Dependencies

Previous TU.

3.15.2.3 Exit Requirements

Feature will be enabled in debug-only mode. Stated functionality verified and checked in.

4. Open Issues

- It is not clear if the saved filter will be a global one or applied only to the folders selected by the user. Consider that the user selects “My Pictures” node, applies some rating and descriptive tags filtering and stores this saved filter. Next time when the user selects this saved filter, will this filter apply to only the items in “My Pictures” folder or to all the items in “All Pictures and Videos”?
- Location of Saved Filters – will it be in the NavPane or the View tab in Ribbon?

5. Future Considerations

In M2, the XML file will have to be tweaked as per the UI. At that time, the XML file will have only one root node and the child nodes will be the filters selected by the user. A sample file is shown below:

```
<?xml version="1.0" ?>
<FilterIntersection name = "FilterBar">
    <FilterDescriptiveTag type = "prefix" query = "foobar" />
    <FilterRating name="Rating" minRating="2" maxRating="4" />
    <FilterFolder pathId="22" recursive="true" />
</FilterIntersection>
```

6. Alternate Solutions

7. Review Feedback