

در [پست قبلی](#) توضیحات کلی درباره WCF Routing Service داده شد و یک مثال را نیز با هم بررسی کردیم. همان طور که در مثال مشاهده شد با استفاده از تعاریف فیلتر در جدول فیلترها توانستیم درخواست‌های مورد نظر را به مقاصد مربوطه اتصال دهیم. در این پست نگاه عمیق‌تری به FilterTable خواهیم داشت.

MessageFilter ها:

با استفاده از این نوع، می‌توان فیلتر مورد نظر را بر روی Message گسترش داد. برای مثال ارزیابی نام فرستنده Message یا حتی نوع عملیات Soap. حتی می‌توانیم فیلترها را با استفاده از And با هم ترکیب نماییم.

FilterType ها

این enum دارای مقادیر زیر است:

Action : با استفاده [ActionMessageFilter](#) فیلتر مورد نظر انجام می‌شود.

And : با استفاده از [StrictAndMessageFilter](#) دو فیلتر مورد نظر را با هم ترکیب می‌کند.

Custom : می‌توان فیلتر مورد نظر را تعریف کرده و این جا فراخوانی نمایید.

MatchAll : با استفاده از [MatchAllMessageFilter](#) تمام فیلترها بررسی خواهند شد.

EndpointAddress : برای فیلتر آدرس درخواست‌های با استفاده از [EndpointAddressMessageFilter](#) مورد استفاده قرار می‌گیرد.

EndpointName : فیلتر با استفاده [EndpointNameMessageFilter](#) بر روی نام Endpoint سرویس مورد نظر انجام می‌گیرد.

FilterData برای تعیین مقادیر مورد نیاز برای FilterType مورد استفاده قرار می‌گیرد.

برای مثال:

```
<filters>
  <filter name="EndpointNameFilter" filterType="EndpointName"
    filterData="calculatorEndpoint"/>
  <filter name="RoundRobinFilter1" filterType="Custom"
    customType="RoutingServiceFilters.RoundRobinMessageFilter,
    RoutingService" filterData="group1"/>
  <filter name="RoundRobinFilter2" filterType="Custom"
    customType="RoutingServiceFilters.RoundRobinMessageFilter,
    RoutingService" filterData="group1"/>
</filters>
```

Filter Table

در واقع مجموعه ای است از اشیای تعریف شده از نوع [FilterTableEntryElement](#) که ارتباط را بین یک فیلتر و مقصد (Endpoint) تعیین می‌نماید. هم چنین امکان تعریف اولویت برای هر کدام از مقصدها یا Endpoint ها وجود دارد. یک مثال:

```
<routing>
  <filters>
    <filter name="AddAction" filterType="Action" filterData="Add" />
    <filter name="SubtractAction" filterType="Action" filterData="Subtract" />
  </filters>
  <filterTables>
    <table name="routingTable1">
      <filters>
        <add filterName="AddAction" endpointName="Addition" />
        <add filterName="SubtractAction" endpointName="Subtraction" />
      </filters>
    </table>
  </filterTables>
</routing>
```

می‌توان برای فیلترها اولویت تعیین کرد. این کار از طریق تنظیم خاصیت Priority امکان پذیر است. در صورت عدم تعیین Priority مقدار پیش فرض صفر خواهد بود.

```
<filterTables>
  <filterTable name="filterTable1">
    <add filterName="EndpointNameFilter" endpointName="regularCalcEndpoint"
      priority="1"/>
    <add filterName="MatchAllMessageFilter" endpointName="defaultCalcEndpoint"
      priority="0"/>
  </filterTable>
</filterTables>
```

در مثال بالا برای یک endpointName مشترک دو فیلتر نوشته شده است با اولویت‌های متفاوت. دو صورتی که اولویت‌ها یکسان باشد با توجه به ترتیب تعریف در filterTable، فیلترها اعمال خواهند شد.

تهیه BackupList

BackupListها این امکانی را در اختیار ما قرار خواهند داد که بتوانیم در صورت عدم موفقیت در عملیات مسیر یابی (برای مثال وقوع CommunicationException) لیستی از مسیرهای جایگزین را تعیین نماییم. در صورت وقوع هر گونه خطا در هنگام فراخوانی سرویس، به جای مواجه شدن با یک استثنا، عملیات مسیر یابی به صورت خودکار به endpointهای تعیین شده در BackupList منتقل خواهد شد.

```
<filterTables>
  <filterTable name="filterTable1">
    <add filterName="MatchAllFilter1" endpointName="Destination"
      backupList="backupEndpointList"/>
  </filterTable>
</filterTables>
<backupLists>
  <backupList name="backupEndpointList">
    <add endpointName="backupServiceQueue" />
    <add endpointName="alternateServiceQueue" />
  </backupList>
</backupLists>
```

در مثال بالا دو endpoint در لیست backup قرار دارد. در صورت وقوع استثنا در Destination عملیات ابتدا به backupServiceQueue منتقل می‌شود و اگر باز هم خطایی وجود داشت نوبت به alternateServiceQueue خواهد رسید.