

```

        this.query = 'select $parent '
        + 'subquery ('
        + '  select $sibling from <#ri> '
        + '  where $sibling <fedora-rels-ext:isMemberOf> $parent '
        + '  and $sibling <fedora-model:hasModel> <fedora:islandora:slideCModel> '
        + '  order by $sibling) '
        + 'from <#ri> '
        + 'where $child <mulgara:is> <fedora:' + this.pid + '> '
        + 'and $child <fedora-rels-ext:isMemberOf> $parent '
        + 'and $parent <fedora-model:hasModel> <fedora:islandora:collectionCModel>;
    }
    else if (this.cmodel == 'islandora-dm:po-document-cmodel') {
        this.query = 'select $object from <#ri> '
        + 'where ($object <fedora-model:hasModel> <fedora:islandora-dm:po-page-cmodel> '
        + 'and $object <fedora-rels-ext:isMemberOf> <fedora:' + this.pid + '>) '
        + 'order by $object';
    }
    else {
        // no query -- pid will be used alone.
    }
    },
    search: function() {
        if (this.query == null) {
            this.results = [this.pid];
        }
        else {
            options = {
                type: this.type,
                lang: this.lang,
                format: this.format,
                query: this.query,
                uid: this.uid
            };
            jQuery.post(this.fedoraUrl + '/risearch', options, this.createCallback(), 'text');
        }
    },
    extractPid: function(riSearchResult) {
        return riSearchResult.replace(/^.*/\//, '');
    },
    createCallback: function() {
        var riSearch = this;
        return function(data, status) {
            var results = [];
            if ('success' == status) {
                var lines = data.split("\n");
                for (i = 0; i < lines.length; i++) {
                    if (i > 0 && lines[i] != '') {
                        results.push(riSearch.extractPid(lines[i]));
                    }
                }
            }
            riSearch.searchCallback(results);
        }
    }
});

/* monkey patch OpenLayers.Layer.OpenURL */
iiv.Viewer.ImageLayer = OpenLayers.Class(OpenLayers.Layer.OpenURL, {
    djatokaUrl: null,
    uid: null,

    /**
     * this implementation is the same as the superclass, except that we use a
     * fedora service as the url base, not djatoka itself
     */
    getURL: function(bounds) {
        bounds = this.adjustBounds(bounds);
        this.calculatePositionAndSize(bounds);
        var z = this.map.getZoom() + this.zoomOffset;

        // uid and djatokaUrl set in createImageLayer
        var path = this.djatokaUrl + '/getRegion?uid=' + this.uid + '&level=' + z
            + '&region=' + this.tilePos.lat + "," + this.tilePos.lon + "," + this.imageSize.h + "," + this.imageSize.w;

        var url = this.url;
        if (url instanceof Array) {
            url = this.selectUrl(path, url);
        }
        return url + path;
    }
});

```

8.1.22 oaiprovider/WEB-INF/classes/proai.properties

```

#
#                               Proai Configuration
#
#####

```