DC3 InputStage Policy Design

updated: Feb 12, 2009

InputStage Policy File Components

1. AdditionalData:
    - type: string
    - description: allows clipboard values to be given names to be used as parameters in pathnames or by Formatters. Form is <name>=<clipboardKey>.
2. RunMode:
    - type: string
    - allowed values: "preprocess", "process", "postprocess"
    - description: specifies when the input should occur.
3. InputItems:
    - type: policy
    - description: list of items to retrieve
    - components:
        1. <itemName>:
            - type: policy
            - description: replace <itemName> with the name you would like to assign to this input item. multiple item policies may be defined within the InputItems policy.
            - components:
                1. Type:
                    - type: string
                    - description: c++ class name. For example: "PropertySet"
                2. PythonType:
                    - type: string
                    - description: Fully qualified python class name. For example: "lsst.daf.base.PropertySet"
                3. StoragePolicy:
                    - type: policy
                    - description: specifies how and where to retrieve this item. Multiple StoragePolicy can be defined for a single item; they are used in sequence.
                    - sub-components:
                        1. Storage:
                            - type: string
                            - allowed values: "BoostStorage", "DbStorage", "DbTsvStorage", "FitsStorage", "XmlStorage"
                            - description: specifies **how** to retrieve the item.
                        2. Location:
                            - type: string
                            - description: specifies **where** to perist the item. Should be a *logical* location, not a *physical* one. Can include %(...) parameters

which will be filled in by the stage using the "AdditionalData" values. If the parameter to be substituted is an integer (not a string representing an integer), it can be formatted using the syntax "%03d(...)". Built in %(...) operands include:

- %(input), %(output), %(scratch), %(update), %(work) directories
- %(dbUrl) database URL
- %(runId), %(sliceId), %(universeSize) pipeline parameters

Sample InputStage Policy File

```
AdditionalData: "exposureId=triggerImageprocEvent.exposureid"
RunMode: "process"
InputItems: {
    InputImage: {
        Type: "MaskedImageF"
        PythonType: "lsst.afw.image.imageLib.MaskedImageF"
        StoragePolicy: {
            Storage: "FitsStorage"
            Location: "/sample/%(exposureId)p_1_MI"
        }
    }
}
```