

Scikit Data Access

Generated by Doxygen 1.8.13

Contents

1	Namespace Index	1
1.1	Packages	1
2	Hierarchical Index	5
2.1	Class Hierarchy	5
3	Class Index	7
3.1	Class List	7
4	File Index	11
4.1	File List	11
5	Namespace Documentation	13
5.1	skdaccess Namespace Reference	13
5.2	skdaccess.astro Namespace Reference	13
5.3	skdaccess.astro.kepler Namespace Reference	13
5.4	skdaccess.astro.kepler.data_fetcher Namespace Reference	14
5.5	skdaccess.astro.spectra Namespace Reference	14
5.6	skdaccess.astro.spectra.stream Namespace Reference	14
5.7	skdaccess.astro.tess Namespace Reference	14
5.8	skdaccess.astro.tess.data Namespace Reference	14
5.9	skdaccess.astro.tess.data.cache Namespace Reference	14
5.10	skdaccess.astro.tess.generic Namespace Reference	15

5.11	skdaccess.astro.tess.generic.cache Namespace Reference	15
5.12	skdaccess.astro.tess.simulated Namespace Reference	15
5.13	skdaccess.astro.tess.simulated.cache Namespace Reference	15
5.14	skdaccess.astro.voyager Namespace Reference	15
5.15	skdaccess.astro.voyager.data_fetcher Namespace Reference	15
5.16	skdaccess.engineering Namespace Reference	16
5.17	skdaccess.engineering.la Namespace Reference	16
5.18	skdaccess.engineering.la.generic Namespace Reference	16
5.19	skdaccess.engineering.la.generic.stream Namespace Reference	16
5.20	skdaccess.engineering.la.traffic_counts Namespace Reference	16
5.21	skdaccess.engineering.la.traffic_counts.stream Namespace Reference	16
5.22	skdaccess.engineering.webcam Namespace Reference	17
5.23	skdaccess.engineering.webcam.mit_sailing Namespace Reference	17
5.24	skdaccess.engineering.webcam.mit_sailing.stream Namespace Reference	17
5.25	skdaccess.finance Namespace Reference	17
5.26	skdaccess.finance.timeseries Namespace Reference	17
5.27	skdaccess.finance.timeseries.stream Namespace Reference	17
5.28	skdaccess.framework Namespace Reference	18
5.29	skdaccess.framework.data_class Namespace Reference	18
5.30	skdaccess.framework.param_class Namespace Reference	18
5.31	skdaccess.geo Namespace Reference	19
5.32	skdaccess.geo.era_interim Namespace Reference	19
5.33	skdaccess.geo.era_interim.cache Namespace Reference	19
5.34	skdaccess.geo.era_interim.cache.data_fetcher Namespace Reference	20
5.35	skdaccess.geo.gldas Namespace Reference	20
5.36	skdaccess.geo.gldas.data_fetcher Namespace Reference	20
5.37	skdaccess.geo.grace Namespace Reference	20
5.38	skdaccess.geo.grace.data_fetcher Namespace Reference	20

5.39	skdaccess.geo.grace.mascon Namespace Reference	20
5.40	skdaccess.geo.grace.mascon.cache Namespace Reference	21
5.41	skdaccess.geo.grace.mascon.cache.data_fetcher Namespace Reference	21
5.42	skdaccess.geo.groundwater Namespace Reference	21
5.43	skdaccess.geo.groundwater.data_fetcher Namespace Reference	21
5.44	skdaccess.geo.imsdnhs Namespace Reference	21
5.45	skdaccess.geo.imsdnhs.data_fetcher Namespace Reference	21
5.46	skdaccess.geo.magnetometer Namespace Reference	22
5.47	skdaccess.geo.magnetometer.data_fetcher Namespace Reference	22
5.48	skdaccess.geo.mahali Namespace Reference	22
5.49	skdaccess.geo.mahali.rinex Namespace Reference	22
5.50	skdaccess.geo.mahali.rinex.data_fetcher Namespace Reference	22
5.51	skdaccess.geo.mahali.rinex.data_wrapper Namespace Reference	22
5.52	skdaccess.geo.mahali.tec Namespace Reference	23
5.53	skdaccess.geo.mahali.tec.data_fetcher Namespace Reference	23
5.54	skdaccess.geo.mahali.temperature Namespace Reference	23
5.55	skdaccess.geo.mahali.temperature.data_fetcher Namespace Reference	23
5.56	skdaccess.geo.modis Namespace Reference	23
5.57	skdaccess.geo.modis.cache Namespace Reference	23
5.58	skdaccess.geo.modis.cache.cloud_mask Namespace Reference	24
5.59	skdaccess.geo.modis.cache.cloud_mask.data_fetcher Namespace Reference	24
5.60	skdaccess.geo.modis.cache.cloud_opacity Namespace Reference	24
5.61	skdaccess.geo.modis.cache.cloud_opacity.data_fetcher Namespace Reference	24
5.62	skdaccess.geo.modis.cache.data_fetcher Namespace Reference	24
5.63	skdaccess.geo.modis.cache.reflectance Namespace Reference	24
5.64	skdaccess.geo.modis.cache.reflectance.data_fetcher Namespace Reference	25
5.65	skdaccess.geo.modis.stream Namespace Reference	25
5.66	skdaccess.geo.modis.stream.cloud_mask Namespace Reference	25

5.67 skdaccess.geo.modis.stream.cloud_mask.data_fetcher Namespace Reference	25
5.68 skdaccess.geo.modis.stream.cloud_opacity Namespace Reference	25
5.69 skdaccess.geo.modis.stream.cloud_opacity.data_fetcher Namespace Reference	25
5.70 skdaccess.geo.modis.stream.data_fetcher Namespace Reference	26
5.71 skdaccess.geo.modis.stream.reflectance Namespace Reference	26
5.72 skdaccess.geo.modis.stream.reflectance.data_fetcher Namespace Reference	26
5.73 skdaccess.geo.ngl_gps Namespace Reference	26
5.74 skdaccess.geo.ngl_gps.data_fetcher Namespace Reference	26
5.75 skdaccess.geo.pbo Namespace Reference	26
5.76 skdaccess.geo.pbo.data_fetcher Namespace Reference	27
5.77 skdaccess.geo.sentinel_1 Namespace Reference	27
5.78 skdaccess.geo.sentinel_1.cache Namespace Reference	27
5.79 skdaccess.geo.sentinel_1.cache.data_fetcher Namespace Reference	27
5.80 skdaccess.geo.srtm Namespace Reference	27
5.81 skdaccess.geo.srtm.cache Namespace Reference	27
5.82 skdaccess.geo.srtm.cache.data_fetcher Namespace Reference	28
5.83 skdaccess.geo.uavsar Namespace Reference	28
5.84 skdaccess.geo.uavsar.cache Namespace Reference	28
5.85 skdaccess.geo.uavsar.cache.data_fetcher Namespace Reference	28
5.86 skdaccess.geo.wyoming_sounding Namespace Reference	28
5.87 skdaccess.geo.wyoming_sounding.cache Namespace Reference	28
5.88 skdaccess.geo.wyoming_sounding.cache.data_fetcher Namespace Reference	29
5.89 skdaccess.geo.wyoming_sounding.stream Namespace Reference	29
5.90 skdaccess.geo.wyoming_sounding.stream.data_fetcher Namespace Reference	29
5.91 skdaccess.planetary Namespace Reference	29
5.92 skdaccess.planetary.ode Namespace Reference	29
5.93 skdaccess.planetary.ode.cache Namespace Reference	29
5.94 skdaccess.planetary.ode.cache.data_fetcher Namespace Reference	30

5.95 skdaccess.solar Namespace Reference	30
5.96 skdaccess.solar.sdo Namespace Reference	30
5.97 skdaccess.solar.sdo.data_fetcher Namespace Reference	30
5.98 skdaccess.utilities Namespace Reference	30
5.99 skdaccess.utilities.file_browser Namespace Reference	31
5.100skdaccess.utilities.file_util Namespace Reference	31
5.100.1 Function Documentation	31
5.100.1.1 openPandasHDFStoreLocking()	31
5.101skdaccess.utilities.grace_util Namespace Reference	31
5.101.1 Function Documentation	32
5.101.1.1 averageDates()	32
5.101.1.2 computeEWD()	32
5.101.1.3 dateMismatch()	33
5.101.1.4 getStartEndDate()	33
5.101.1.5 readTellusData()	33
5.102skdaccess.utilities.gw_util Namespace Reference	34
5.102.1 Function Documentation	34
5.102.1.1 combine_water_heights()	35
5.103skdaccess.utilities.image_util Namespace Reference	36
5.103.1 Function Documentation	36
5.103.1.1 convertBinCentersToEdges()	36
5.103.1.2 getExtentsFromCentersPlateCarree()	37
5.103.1.3 getGeoTransform()	37
5.103.1.4 SplineGeolocation()	38
5.103.2 Variable Documentation	38
5.103.2.1 lat_spline	38
5.103.2.2 lon_spline	38
5.103.2.3 x_offset	38

5.103.2.4 x_spline	38
5.103.2.5 y_offset	38
5.103.2.6 y_spline	39
5.104skdaccess.utilities.kepler_util Namespace Reference	39
5.104.1 Function Documentation	39
5.104.1.1 normalize()	39
5.105skdaccess.utilities.mahali_util Namespace Reference	39
5.105.1 Function Documentation	40
5.105.1.1 convert_date()	40
5.105.1.2 parselonoFile()	40
5.106skdaccess.utilities.modis_util Namespace Reference	40
5.106.1 Function Documentation	41
5.106.1.1 calibrateModis()	41
5.106.1.2 checkBit()	41
5.106.1.3 createGrid()	42
5.106.1.4 getFileIds()	42
5.106.1.5 getFileURLs()	43
5.106.1.6 getImageType()	43
5.106.1.7 getModisData()	44
5.106.1.8 readMODISData()	44
5.106.1.9 rescale()	45
5.107skdaccess.utilities.ode_util Namespace Reference	45
5.107.1 Function Documentation	46
5.107.1.1 correct_CRISM_label()	46
5.107.1.2 correct_file_name_case_in_label()	46
5.107.1.3 correct_label_file()	46
5.107.1.4 get_files_urls()	47
5.107.1.5 get_query_url()	47

5.107.1.6 <code>get_raster_array()</code>	47
5.107.1.7 <code>get_raster_extent()</code>	48
5.107.1.8 <code>query_files_urls()</code>	48
5.107.1.9 <code>query_yes_no()</code>	49
5.108skdaccess.utilities.pbo_util Namespace Reference	49
5.108.1 Function Documentation	49
5.108.1.1 <code>getLatLonRange()</code>	50
5.108.1.2 <code>getROIstations()</code>	50
5.108.1.3 <code>getStationCoords()</code>	51
5.108.1.4 <code>nostab_sys()</code>	51
5.108.1.5 <code>propagateErrors()</code>	52
5.108.1.6 <code>removeAntennaOffset()</code>	52
5.108.1.7 <code>stab_sys()</code>	53
5.109skdaccess.utilities.sentinel_1_util Namespace Reference	53
5.109.1 Function Documentation	53
5.109.1.1 <code>parseSatelliteData()</code>	54
5.110skdaccess.utilities.sounding_util Namespace Reference	54
5.110.1 Function Documentation	54
5.110.1.1 <code>generateQueries()</code>	54
5.111skdaccess.utilities.srtm_util Namespace Reference	55
5.111.1 Function Documentation	55
5.111.1.1 <code>getSRTMData()</code>	55
5.111.1.2 <code>getSRTMLatLon()</code>	56
5.111.1.3 <code>merge_srtm_tiles()</code>	56
5.112skdaccess.utilities.support Namespace Reference	56
5.112.1 Function Documentation	57
5.112.1.1 <code>convertToStr()</code>	57
5.112.1.2 <code>join_string()</code>	57

5.112.1.3 progress_bar()	57
5.112.1.4 retrieveCommonDatesHDF()	58
5.113skdaccess.utilities.tess_utils Namespace Reference	58
5.113.1 Function Documentation	58
5.113.1.1 parseTessData()	58
5.114skdaccess.utilities.uavsar_util Namespace Reference	59
5.114.1 Function Documentation	59
5.114.1.1 readUAVSARMetadata()	59
5.115terminal_groundwater_example Namespace Reference	59
5.115.1 Variable Documentation	60
5.115.1.1 color	60
5.115.1.2 data_1	60
5.115.1.3 data_2	60
5.115.1.4 datalt	60
5.115.1.5 fullIDF	60
5.115.1.6 fullIDW	61
5.115.1.7 label_1	61
5.115.1.8 label_2	61
5.115.1.9 meta_data	61

6	Class Documentation	63
6.1	skdaccess.utilities.image_util.AffineGlobalCoords Class Reference	63
6.1.1	Detailed Description	63
6.1.2	Constructor & Destructor Documentation	63
6.1.2.1	<code>__init__()</code>	64
6.1.3	Member Function Documentation	64
6.1.3.1	<code>getPixelYX()</code>	64
6.1.3.2	<code>getProjectedYX()</code>	64
6.2	skdaccess.framework.param_class.AutoList Class Reference	65
6.2.1	Detailed Description	66
6.2.2	Constructor & Destructor Documentation	66
6.2.2.1	<code>__init__()</code>	66
6.2.3	Member Function Documentation	66
6.2.3.1	<code>__call__()</code>	66
6.2.3.2	<code>__getitem__()</code>	67
6.2.3.3	<code>__len__()</code>	67
6.2.3.4	<code>__setitem__()</code>	67
6.2.3.5	<code>__str__()</code>	68
6.2.3.6	<code>getAllOptions()</code>	68
6.2.3.7	<code>perturb()</code>	68
6.2.3.8	<code>reset()</code>	68
6.2.3.9	<code>val()</code>	69
6.2.4	Member Data Documentation	69
6.2.4.1	<code>val_init</code>	69
6.2.4.2	<code>val_list</code>	69
6.3	skdaccess.framework.param_class.AutoListCycle Class Reference	69
6.3.1	Detailed Description	70
6.3.2	Constructor & Destructor Documentation	70

6.3.2.1	<code>__init__()</code>	70
6.3.3	Member Function Documentation	71
6.3.3.1	<code>__call__()</code>	71
6.3.3.2	<code>__getitem__()</code>	71
6.3.3.3	<code>__len__()</code>	71
6.3.3.4	<code>__setitem__()</code>	72
6.3.3.5	<code>__str__()</code>	72
6.3.3.6	<code>getAllOptions()</code>	72
6.3.3.7	<code>perturb()</code>	73
6.3.3.8	<code>reset()</code>	73
6.3.3.9	<code>val()</code>	73
6.3.4	Member Data Documentation	73
6.3.4.1	<code>index</code>	73
6.3.4.2	<code>list_val_list</code>	73
6.3.4.3	<code>val_init</code>	74
6.3.4.4	<code>val_list</code>	74
6.4	<code>skdaccess.framework.param_class.AutoListPermute</code> Class Reference	74
6.4.1	Detailed Description	75
6.4.2	Member Function Documentation	75
6.4.2.1	<code>__call__()</code>	75
6.4.2.2	<code>__getitem__()</code>	75
6.4.2.3	<code>__len__()</code>	76
6.4.2.4	<code>__setitem__()</code>	76
6.4.2.5	<code>__str__()</code>	76
6.4.2.6	<code>getAllOptions()</code>	77
6.4.2.7	<code>perturb()</code>	77
6.4.2.8	<code>reset()</code>	77
6.4.2.9	<code>val()</code>	77

6.4.3	Member Data Documentation	77
6.4.3.1	val_init	78
6.4.3.2	val_list	78
6.5	skdaccess.framework.param_class.AutoListRemove Class Reference	78
6.5.1	Detailed Description	79
6.5.2	Constructor & Destructor Documentation	79
6.5.2.1	__init__()	79
6.5.3	Member Function Documentation	79
6.5.3.1	__call__()	79
6.5.3.2	__getitem__()	80
6.5.3.3	__len__()	80
6.5.3.4	__setitem__()	80
6.5.3.5	__str__()	81
6.5.3.6	getAllOptions()	81
6.5.3.7	perturb()	81
6.5.3.8	reset()	81
6.5.3.9	val()	82
6.5.4	Member Data Documentation	82
6.5.4.1	n	82
6.5.4.2	val_init	82
6.5.4.3	val_list	82
6.6	skdaccess.framework.param_class.AutoListSubset Class Reference	82
6.6.1	Detailed Description	83
6.6.2	Member Function Documentation	83
6.6.2.1	__call__()	83
6.6.2.2	__getitem__()	84
6.6.2.3	__len__()	84
6.6.2.4	__setitem__()	84

6.6.2.5	<code>__str__()</code>	85
6.6.2.6	<code>getAllOptions()</code>	85
6.6.2.7	<code>perturb()</code>	85
6.6.2.8	<code>reset()</code>	85
6.6.2.9	<code>val()</code>	86
6.6.3	Member Data Documentation	86
6.6.3.1	<code>val_init</code>	86
6.6.3.2	<code>val_list</code>	86
6.7	<code>skdaccess.framework.param_class.AutoParam</code> Class Reference	86
6.7.1	Detailed Description	87
6.7.2	Constructor & Destructor Documentation	87
6.7.2.1	<code>__init__()</code>	87
6.7.3	Member Function Documentation	88
6.7.3.1	<code>__call__()</code>	88
6.7.3.2	<code>__str__()</code>	88
6.7.3.3	<code>perturb()</code>	88
6.7.3.4	<code>reset()</code>	88
6.7.4	Member Data Documentation	89
6.7.4.1	<code>val</code>	89
6.7.4.2	<code>val_init</code>	89
6.8	<code>skdaccess.framework.param_class.AutoParamList</code> Class Reference	89
6.8.1	Detailed Description	90
6.8.2	Constructor & Destructor Documentation	90
6.8.2.1	<code>__init__()</code>	90
6.8.3	Member Function Documentation	90
6.8.3.1	<code>__call__()</code>	90
6.8.3.2	<code>__str__()</code>	91
6.8.3.3	<code>perturb()</code>	91

6.8.3.4	reset()	91
6.8.4	Member Data Documentation	91
6.8.4.1	val	91
6.8.4.2	val_init	91
6.8.4.3	val_list	92
6.9	skdaccess.framework.param_class.AutoParamListCycle Class Reference	92
6.9.1	Detailed Description	92
6.9.2	Constructor & Destructor Documentation	93
6.9.2.1	__init__()	93
6.9.3	Member Function Documentation	93
6.9.3.1	__call__()	93
6.9.3.2	__str__()	93
6.9.3.3	perturb()	94
6.9.3.4	reset()	94
6.9.4	Member Data Documentation	94
6.9.4.1	current_index	94
6.9.4.2	val	94
6.9.4.3	val_init	94
6.9.4.4	val_list	94
6.10	skdaccess.framework.param_class.AutoParamMinMax Class Reference	95
6.10.1	Detailed Description	95
6.10.2	Constructor & Destructor Documentation	95
6.10.2.1	__init__()	96
6.10.3	Member Function Documentation	96
6.10.3.1	__call__()	96
6.10.3.2	__str__()	96
6.10.3.3	perturb()	97
6.10.3.4	reset()	97

6.10.4	Member Data Documentation	97
6.10.4.1	decimals	97
6.10.4.2	n	97
6.10.4.3	n_max	97
6.10.4.4	val	97
6.10.4.5	val_init	98
6.10.4.6	val_max	98
6.10.4.7	val_min	98
6.11	skdaccess.solar.sdo.DataFetcher Class Reference	98
6.11.1	Detailed Description	99
6.11.2	Constructor & Destructor Documentation	99
6.11.2.1	__init__()	99
6.11.3	Member Function Documentation	100
6.11.3.1	__str__()	100
6.11.3.2	getConfig()	100
6.11.3.3	getConfigItem()	100
6.11.3.4	getMetadata()	101
6.11.3.5	multirun_enabled()	101
6.11.3.6	output()	101
6.11.3.7	perturb()	102
6.11.3.8	reset()	102
6.11.3.9	retrieveOnlineData()	102
6.11.3.10	verbose_print()	102
6.11.3.11	writeConfig()	103
6.11.3.12	writeConfigItem()	103
6.11.4	Member Data Documentation	103
6.11.4.1	ap_paramList	104
6.11.4.2	verbose	104

6.12	skdaccess.geo.era_interim.cache.DataFetcher Class Reference	104
6.12.1	Detailed Description	105
6.12.2	Constructor & Destructor Documentation	105
6.12.2.1	__init__()	106
6.12.3	Member Function Documentation	106
6.12.3.1	__str__()	106
6.12.3.2	cacheData()	106
6.12.3.3	checkIfDataExists()	107
6.12.3.4	getConfig()	107
6.12.3.5	getConfigItem()	108
6.12.3.6	getDataLocation()	108
6.12.3.7	getHDFSStorage()	108
6.12.3.8	getMetadata()	109
6.12.3.9	multirun_enabled()	109
6.12.3.10	output()	109
6.12.3.11	perturb()	110
6.12.3.12	reset()	110
6.12.3.13	setDataLocation()	110
6.12.3.14	verbose_print()	110
6.12.3.15	writeConfig()	111
6.12.3.16	writeConfigItem()	111
6.12.4	Member Data Documentation	111
6.12.4.1	ap_paramList	112
6.12.4.2	data_names	112
6.12.4.3	date_list	112
6.12.4.4	password	112
6.12.4.5	username	112
6.12.4.6	verbose	112

6.13	skdaccess.astro.tess.simulated.cache.DataFetcher Class Reference	113
6.13.1	Detailed Description	113
6.13.2	Constructor & Destructor Documentation	113
6.13.2.1	<code>__init__()</code>	113
6.13.3	Member Function Documentation	114
6.13.3.1	<code>generateURLFromTID()</code>	114
6.13.3.2	<code>getTargetInformation()</code>	114
6.13.4	Member Data Documentation	114
6.13.4.1	<code>end_url</code>	114
6.13.4.2	<code>start_url</code>	115
6.14	skdaccess.geo.modis.stream.cloud_opacity.DataFetcher Class Reference	115
6.14.1	Detailed Description	115
6.14.2	Constructor & Destructor Documentation	115
6.14.2.1	<code>__init__()</code>	115
6.15	skdaccess.astro.tess.generic.cache.DataFetcher Class Reference	116
6.15.1	Detailed Description	117
6.15.2	Constructor & Destructor Documentation	117
6.15.2.1	<code>__init__()</code>	118
6.15.3	Member Function Documentation	118
6.15.3.1	<code>__str__()</code>	118
6.15.3.2	<code>cacheData()</code>	118
6.15.3.3	<code>checkIfDataExists()</code>	119
6.15.3.4	<code>generateURLFromTID()</code>	119
6.15.3.5	<code>getConfig()</code>	120
6.15.3.6	<code>getConfigItem()</code>	120
6.15.3.7	<code>getDataLocation()</code>	120
6.15.3.8	<code>getHDFSStorage()</code>	121
6.15.3.9	<code>getMetadata()</code>	121

6.15.3.10	getTargetInformation()	122
6.15.3.11	multirun_enabled()	122
6.15.3.12	output()	122
6.15.3.13	perturb()	122
6.15.3.14	reset()	123
6.15.3.15	setDataLocation()	123
6.15.3.16	verbose_print()	123
6.15.3.17	writeConfig()	123
6.15.3.18	writeConfigItem()	124
6.15.4	Member Data Documentation	124
6.15.4.1	ap_paramList	124
6.15.4.2	toi_information	124
6.15.4.3	verbose	125
6.16	skdaccess.geo.groundwater.DataFetcher Class Reference	125
6.16.1	Detailed Description	126
6.16.2	Constructor & Destructor Documentation	126
6.16.2.1	__init__()	126
6.16.3	Member Function Documentation	127
6.16.3.1	__str__()	127
6.16.3.2	downloadFullDataset()	127
6.16.3.3	getConfig()	128
6.16.3.4	getConfigItem()	128
6.16.3.5	getDataLocation()	128
6.16.3.6	getMetadata()	129
6.16.3.7	getStationMetadata()	129
6.16.3.8	multirun_enabled()	129
6.16.3.9	output()	130
6.16.3.10	perturb()	130

6.16.3.11	reset()	130
6.16.3.12	setDataLocation()	130
6.16.3.13	verbose_print()	131
6.16.3.14	writeConfig()	131
6.16.3.15	writeConfigItem()	131
6.16.4	Member Data Documentation	132
6.16.4.1	ap_paramList	132
6.16.4.2	cutoff	132
6.16.4.3	end_date	132
6.16.4.4	start_date	132
6.16.4.5	verbose	132
6.17	skdaccess.astro.kepler.DataFetcher Class Reference	133
6.17.1	Detailed Description	134
6.17.2	Constructor & Destructor Documentation	134
6.17.2.1	__init__()	134
6.17.3	Member Function Documentation	135
6.17.3.1	__str__()	135
6.17.3.2	cacheData() [1/2]	135
6.17.3.3	cacheData() [2/2]	135
6.17.3.4	checkIfDataExists()	136
6.17.3.5	downloadKeplerData()	136
6.17.3.6	getConfig()	137
6.17.3.7	getConfigItem()	137
6.17.3.8	getDataLocation()	137
6.17.3.9	getHDFStorage()	138
6.17.3.10	getMetadata()	138
6.17.3.11	multirun_enabled()	139
6.17.3.12	output()	139

6.17.3.13 perturb()	139
6.17.3.14 reset()	139
6.17.3.15 setDataLocation()	139
6.17.3.16 verbose_print()	140
6.17.3.17 writeConfig()	140
6.17.3.18 writeConfigItem()	140
6.17.4 Member Data Documentation	141
6.17.4.1 ap_paramList	141
6.17.4.2 quarter_list	141
6.17.4.3 verbose	141
6.18 skdaccess.engineering.la.generic.stream.DataFetcher Class Reference	141
6.18.1 Detailed Description	142
6.18.2 Constructor & Destructor Documentation	143
6.18.2.1 __init__()	143
6.18.3 Member Function Documentation	143
6.18.3.1 __str__()	143
6.18.3.2 getConfig()	143
6.18.3.3 getConfigItem()	144
6.18.3.4 getMetadata()	144
6.18.3.5 multirun_enabled()	144
6.18.3.6 output()	145
6.18.3.7 perturb()	145
6.18.3.8 reset()	145
6.18.3.9 retrieveOnlineData()	145
6.18.3.10 verbose_print()	146
6.18.3.11 writeConfig()	146
6.18.3.12 writeConfigItem()	146
6.18.4 Member Data Documentation	147

6.18.4.1	ap_paramList	147
6.18.4.2	app_token	147
6.18.4.3	base_url	147
6.18.4.4	base_url_and_endpoint	147
6.18.4.5	label	147
6.18.4.6	pandas_kwargs	148
6.18.4.7	parameters	148
6.18.4.8	verbose	148
6.19	skdaccess.geo.wyoming_sounding.stream.DataFetcher Class Reference	148
6.19.1	Detailed Description	149
6.19.2	Constructor & Destructor Documentation	150
6.19.2.1	__init__()	150
6.19.3	Member Function Documentation	150
6.19.3.1	__str__()	150
6.19.3.2	getConfig()	151
6.19.3.3	getConfigItem()	151
6.19.3.4	getMetadata()	151
6.19.3.5	multirun_enabled()	152
6.19.3.6	output() [1/2]	152
6.19.3.7	output() [2/2]	152
6.19.3.8	perturb()	152
6.19.3.9	reset()	153
6.19.3.10	retrieveOnlineData()	153
6.19.3.11	verbose_print()	153
6.19.3.12	writeConfig()	153
6.19.3.13	writeConfigItem()	154
6.19.4	Member Data Documentation	154
6.19.4.1	ap_paramList	154

6.19.4.2	day_end	154
6.19.4.3	day_start	155
6.19.4.4	end_hour	155
6.19.4.5	month_list	155
6.19.4.6	start_hour	155
6.19.4.7	station_number	155
6.19.4.8	verbose	155
6.19.4.9	year_list	155
6.20	skdaccess.geo.srtm.cache.DataFetcher Class Reference	156
6.20.1	Detailed Description	157
6.20.2	Constructor & Destructor Documentation	157
6.20.2.1	__init__()	157
6.20.3	Member Function Documentation	158
6.20.3.1	__str__()	158
6.20.3.2	cacheData()	158
6.20.3.3	checkIfDataExists()	159
6.20.3.4	getConfig()	159
6.20.3.5	getConfigItem()	159
6.20.3.6	getDataLocation()	160
6.20.3.7	getHDFStorage()	160
6.20.3.8	getMetadata()	161
6.20.3.9	multirun_enabled()	161
6.20.3.10	output()	161
6.20.3.11	perturb()	161
6.20.3.12	reset()	162
6.20.3.13	setDataLocation()	162
6.20.3.14	verbose_print()	162
6.20.3.15	writeConfig()	162

6.20.3.16 writeConfigItem()	163
6.20.4 Member Data Documentation	163
6.20.4.1 ap_paramList	163
6.20.4.2 arcsecond_sampling	163
6.20.4.3 lat_tile_end	164
6.20.4.4 lat_tile_start	164
6.20.4.5 lon_tile_end	164
6.20.4.6 lon_tile_start	164
6.20.4.7 mask_water	164
6.20.4.8 password	165
6.20.4.9 store_geolocation_grids	165
6.20.4.10 username	165
6.20.4.11 verbose	165
6.21 skdaccess.geo.modis.cache.reflectance.DataFetcher Class Reference	165
6.21.1 Detailed Description	166
6.21.2 Constructor & Destructor Documentation	166
6.21.2.1 __init__()	166
6.22 skdaccess.geo.mahali.temperature.DataFetcher Class Reference	166
6.22.1 Detailed Description	168
6.22.2 Constructor & Destructor Documentation	168
6.22.2.1 __init__()	168
6.22.3 Member Function Documentation	168
6.22.3.1 __str__()	168
6.22.3.2 getConfig()	168
6.22.3.3 getConfigItem()	169
6.22.3.4 getMetadata()	169
6.22.3.5 multirun_enabled()	169
6.22.3.6 output()	170

6.22.3.7	perturb()	170
6.22.3.8	reset()	170
6.22.3.9	retrieveOnlineData()	170
6.22.3.10	verbose_print()	171
6.22.3.11	writeConfig()	171
6.22.3.12	writeConfigItem()	171
6.22.4	Member Data Documentation	172
6.22.4.1	ap_paramList	172
6.22.4.2	end_date	172
6.22.4.3	start_date	172
6.22.4.4	verbose	172
6.23	skdaccess.geo.modis.cache.cloud_opacity.DataFetcher Class Reference	173
6.23.1	Detailed Description	173
6.23.2	Constructor & Destructor Documentation	173
6.23.2.1	__init__()	173
6.24	skdaccess.geo.modis.stream.cloud_mask.DataFetcher Class Reference	174
6.24.1	Detailed Description	174
6.24.2	Constructor & Destructor Documentation	174
6.24.2.1	__init__()	174
6.25	skdaccess.finance.timeseries.stream.DataFetcher Class Reference	175
6.25.1	Detailed Description	176
6.25.2	Constructor & Destructor Documentation	176
6.25.2.1	__init__()	176
6.25.3	Member Function Documentation	177
6.25.3.1	__str__()	177
6.25.3.2	getConfig()	177
6.25.3.3	getConfigItem()	177
6.25.3.4	getMetadata()	178

6.25.3.5	multirun_enabled()	178
6.25.3.6	output()	178
6.25.3.7	perturb()	179
6.25.3.8	reset()	179
6.25.3.9	retrieveOnlineData()	179
6.25.3.10	verbose_print()	179
6.25.3.11	writeConfig()	180
6.25.3.12	writeConfigItem()	180
6.25.4	Member Data Documentation	180
6.25.4.1	ap_paramList	181
6.25.4.2	data_type	181
6.25.4.3	end_date	181
6.25.4.4	interval	181
6.25.4.5	possible_data_types	181
6.25.4.6	possible_intervals	181
6.25.4.7	start_date	181
6.25.4.8	verbose	182
6.26	skdaccess.geo.modis.cache.cloud_mask.DataFetcher Class Reference	182
6.26.1	Detailed Description	182
6.26.2	Constructor & Destructor Documentation	182
6.26.2.1	__init__()	182
6.27	skdaccess.geo.mahali.rinex.DataFetcher Class Reference	183
6.27.1	Detailed Description	184
6.27.2	Constructor & Destructor Documentation	184
6.27.2.1	__init__()	185
6.27.3	Member Function Documentation	185
6.27.3.1	__str__()	185
6.27.3.2	cacheData() [1/2]	185

6.27.3.3	cacheData() [2/2]	186
6.27.3.4	checkIfDataExists()	186
6.27.3.5	getConfig()	187
6.27.3.6	getConfigItem()	187
6.27.3.7	getDataLocation()	187
6.27.3.8	getHDFSStorage()	188
6.27.3.9	getMetadata()	188
6.27.3.10	multirun_enabled()	189
6.27.3.11	output()	189
6.27.3.12	perturb()	189
6.27.3.13	reset()	189
6.27.3.14	setDataLocation()	189
6.27.3.15	verbose_print()	190
6.27.3.16	writeConfig()	190
6.27.3.17	writeConfigItem()	190
6.27.4	Member Data Documentation	191
6.27.4.1	ap_paramList	191
6.27.4.2	date_range	191
6.27.4.3	end_date	191
6.27.4.4	generate_links	191
6.27.4.5	start_date	191
6.27.4.6	verbose	192
6.28	skdaccess.geo.gldas.DataFetcher Class Reference	192
6.28.1	Detailed Description	193
6.28.2	Constructor & Destructor Documentation	193
6.28.2.1	__init__()	193
6.28.3	Member Function Documentation	194
6.28.3.1	__str__()	194

6.28.3.2	downloadFullDataset()	194
6.28.3.3	getConfig()	194
6.28.3.4	getConfigItem()	195
6.28.3.5	getDataLocation()	195
6.28.3.6	getMetadata()	195
6.28.3.7	multirun_enabled()	196
6.28.3.8	output()	196
6.28.3.9	perturb()	196
6.28.3.10	reset()	196
6.28.3.11	setDataLocation()	196
6.28.3.12	verbose_print()	197
6.28.3.13	writeConfig()	197
6.28.3.14	writeConfigItem()	197
6.28.4	Member Data Documentation	198
6.28.4.1	ap_paramList	198
6.28.4.2	end_date	198
6.28.4.3	resample	198
6.28.4.4	start_date	198
6.28.4.5	verbose	198
6.29	skdaccess.engineering.la.traffic_counts.stream.DataFetcher Class Reference	199
6.29.1	Detailed Description	199
6.29.2	Constructor & Destructor Documentation	199
6.29.2.1	__init__()	199
6.30	skdaccess.geo.mahali.tec.DataFetcher Class Reference	200
6.30.1	Detailed Description	201
6.30.2	Constructor & Destructor Documentation	201
6.30.2.1	__init__()	201
6.30.3	Member Function Documentation	202

6.30.3.1	<code>__str__()</code>	202
6.30.3.2	<code>cacheData()</code>	202
6.30.3.3	<code>checkIfDataExists()</code>	203
6.30.3.4	<code>getConfig()</code>	203
6.30.3.5	<code>getConfigItem()</code>	203
6.30.3.6	<code>getDataLocation()</code>	204
6.30.3.7	<code>getHDFSStorage()</code>	204
6.30.3.8	<code>getMetadata()</code>	204
6.30.3.9	<code>multirun_enabled()</code>	205
6.30.3.10	<code>output()</code>	205
6.30.3.11	<code>perturb()</code>	205
6.30.3.12	<code>reset()</code>	205
6.30.3.13	<code>setDataLocation()</code>	205
6.30.3.14	<code>verbose_print()</code>	206
6.30.3.15	<code>writeConfig()</code>	206
6.30.3.16	<code>writeConfigItem()</code>	206
6.30.4	Member Data Documentation	207
6.30.4.1	<code>ap_paramList</code>	207
6.30.4.2	<code>date_range</code>	207
6.30.4.3	<code>end_date</code>	207
6.30.4.4	<code>start_date</code>	207
6.30.4.5	<code>verbose</code>	207
6.31	<code>skdaccess.geo.magnetometer.DataFetcher</code> Class Reference	208
6.31.1	Detailed Description	209
6.31.2	Constructor & Destructor Documentation	209
6.31.2.1	<code>__init__()</code>	209
6.31.3	Member Function Documentation	209
6.31.3.1	<code>__str__()</code>	210

6.31.3.2	getConfig()	210
6.31.3.3	getConfigItem()	210
6.31.3.4	getDataMetadata()	210
6.31.3.5	getMetadata()	211
6.31.3.6	multirun_enabled()	211
6.31.3.7	output()	211
6.31.3.8	perturb()	212
6.31.3.9	reset()	212
6.31.3.10	retrieveOnlineData()	212
6.31.3.11	verbose_print()	212
6.31.3.12	writeConfig()	213
6.31.3.13	writeConfigItem()	213
6.31.4	Member Data Documentation	213
6.31.4.1	ap_paramList	214
6.31.4.2	channels	214
6.31.4.3	data_type	214
6.31.4.4	end_time	214
6.31.4.5	interval	214
6.31.4.6	start_time	214
6.31.4.7	verbose	214
6.32	skdaccess.geo.uavsar.cache.DataFetcher Class Reference	215
6.32.1	Detailed Description	216
6.32.2	Constructor & Destructor Documentation	216
6.32.2.1	__init__()	216
6.32.3	Member Function Documentation	217
6.32.3.1	__str__()	217
6.32.3.2	cacheData()	217
6.32.3.3	checkIfDataExists()	218

6.32.3.4	getConfig()	218
6.32.3.5	getConfigItem()	218
6.32.3.6	getDataLocation()	219
6.32.3.7	getHDFSStorage()	219
6.32.3.8	getMetadata()	219
6.32.3.9	multirun_enabled()	220
6.32.3.10	output()	220
6.32.3.11	perturb()	220
6.32.3.12	reset()	220
6.32.3.13	setDataLocation()	220
6.32.3.14	verbose_print()	221
6.32.3.15	writeConfig()	221
6.32.3.16	writeConfigItem()	221
6.32.4	Member Data Documentation	222
6.32.4.1	ap_paramList	222
6.32.4.2	llh_url	222
6.32.4.3	memmap	222
6.32.4.4	metadata_url_list	222
6.32.4.5	slc_url_list	222
6.32.4.6	verbose	223
6.33	skdaccess.geo.ngl_gps.DataFetcher Class Reference	223
6.33.1	Detailed Description	224
6.33.2	Constructor & Destructor Documentation	224
6.33.2.1	__init__()	224
6.33.3	Member Function Documentation	225
6.33.3.1	__str__()	225
6.33.3.2	downloadFullDataset()	225
6.33.3.3	getAntennaLogs()	226

6.33.3.4	getConfig()	226
6.33.3.5	getConfigItem()	226
6.33.3.6	getDataLocation()	227
6.33.3.7	getMetadata()	227
6.33.3.8	getStationMetadata()	227
6.33.3.9	multirun_enabled()	228
6.33.3.10	output()	228
6.33.3.11	perturb()	228
6.33.3.12	reset()	228
6.33.3.13	setDataLocation()	228
6.33.3.14	verbose_print()	229
6.33.3.15	writeConfig()	229
6.33.3.16	writeConfigItem()	229
6.33.4	Member Data Documentation	230
6.33.4.1	ap_paramList	230
6.33.4.2	data_type	230
6.33.4.3	end_date	230
6.33.4.4	lat_range	230
6.33.4.5	lon_range	230
6.33.4.6	mdyratio	231
6.33.4.7	start_date	231
6.33.4.8	verbose	231
6.34	skdaccess.geo.modis.cache.DataFetcher Class Reference	231
6.34.1	Detailed Description	233
6.34.2	Constructor & Destructor Documentation	233
6.34.2.1	__init__()	233
6.34.3	Member Function Documentation	234
6.34.3.1	__str__()	234

6.34.3.2	cacheData() [1/2]	234
6.34.3.3	cacheData() [2/2]	234
6.34.3.4	checkIfDataExists()	235
6.34.3.5	find_data()	236
6.34.3.6	getConfig()	236
6.34.3.7	getConfigItem()	236
6.34.3.8	getDataLocation()	237
6.34.3.9	getHDFSStorage()	237
6.34.3.10	getMetadata()	237
6.34.3.11	multirun_enabled()	238
6.34.3.12	output()	238
6.34.3.13	perturb()	238
6.34.3.14	reset()	238
6.34.3.15	setDataLocation()	238
6.34.3.16	verbose_print()	239
6.34.3.17	writeConfig()	239
6.34.3.18	writeConfigItem()	239
6.34.4	Member Data Documentation	240
6.34.4.1	ap_paramList	240
6.34.4.2	daynightboth	240
6.34.4.3	end_date	240
6.34.4.4	grid	240
6.34.4.5	grid_fill	240
6.34.4.6	modis_id	241
6.34.4.7	modis_identifier	241
6.34.4.8	modis_platform	241
6.34.4.9	start_date	241
6.34.4.10	use_long_name	241

6.34.4.11	<code>variable_list</code>	241
6.34.4.12	<code>verbose</code>	241
6.35	<code>skdaccess.geo.modis.stream.reflectance.DataFetcher</code> Class Reference	242
6.35.1	Detailed Description	242
6.35.2	Constructor & Destructor Documentation	242
6.35.2.1	<code>__init__()</code>	242
6.36	<code>skdaccess.geo.wyoming_sounding.cache.DataFetcher</code> Class Reference	243
6.36.1	Detailed Description	244
6.36.2	Constructor & Destructor Documentation	244
6.36.2.1	<code>__init__()</code>	244
6.36.3	Member Function Documentation	245
6.36.3.1	<code>__str__()</code>	245
6.36.3.2	<code>cacheData()</code>	245
6.36.3.3	<code>checkIfDataExists()</code>	246
6.36.3.4	<code>getConfig()</code>	246
6.36.3.5	<code>getConfigItem()</code>	246
6.36.3.6	<code>getDataLocation()</code>	247
6.36.3.7	<code>getHDFStorage()</code>	247
6.36.3.8	<code>getMetadata()</code>	248
6.36.3.9	<code>multirun_enabled()</code>	248
6.36.3.10	<code>output()</code>	248
6.36.3.11	<code>perturb()</code>	248
6.36.3.12	<code>reset()</code>	249
6.36.3.13	<code>setDataLocation()</code>	249
6.36.3.14	<code>verbose_print()</code>	249
6.36.3.15	<code>writeConfig()</code>	249
6.36.3.16	<code>writeConfigItem()</code>	250
6.36.4	Member Data Documentation	250

6.36.4.1	ap_paramList	250
6.36.4.2	day_end	250
6.36.4.3	day_start	251
6.36.4.4	end_hour	251
6.36.4.5	month_list	251
6.36.4.6	start_hour	251
6.36.4.7	station_number	251
6.36.4.8	verbose	251
6.36.4.9	year_list	251
6.37	skdaccess.geo.pbo.DataFetcher Class Reference	252
6.37.1	Detailed Description	253
6.37.2	Constructor & Destructor Documentation	253
6.37.2.1	__init__()	254
6.37.3	Member Function Documentation	255
6.37.3.1	__str__()	255
6.37.3.2	downloadFullDataset()	255
6.37.3.3	getAntennaLogs()	256
6.37.3.4	getConfig()	256
6.37.3.5	getConfigItem()	256
6.37.3.6	getDataLocation()	257
6.37.3.7	getInfo()	257
6.37.3.8	getMetadata()	257
6.37.3.9	getStationMetadata()	258
6.37.3.10	multirun_enabled()	258
6.37.3.11	output()	258
6.37.3.12	perturb()	258
6.37.3.13	reset()	259
6.37.3.14	setDataLocation()	259

6.37.3.15	setStationList()	259
6.37.3.16	verbose_print()	259
6.37.3.17	writeConfig()	260
6.37.3.18	writeConfigItem()	260
6.37.4	Member Data Documentation	261
6.37.4.1	antenna_info	261
6.37.4.2	ap_paramList	261
6.37.4.3	default_columns	261
6.37.4.4	default_error_columns	261
6.37.4.5	index_date_only	261
6.37.4.6	meta_data	261
6.37.4.7	station_list	262
6.37.4.8	use_progress_bar	262
6.37.4.9	verbose	262
6.38	skdaccess.geo.grace.DataFetcher Class Reference	262
6.38.1	Detailed Description	263
6.38.2	Constructor & Destructor Documentation	264
6.38.2.1	__init__()	264
6.38.3	Member Function Documentation	264
6.38.3.1	__str__()	264
6.38.3.2	downloadFullDataset()	264
6.38.3.3	getConfig()	265
6.38.3.4	getConfigItem()	265
6.38.3.5	getDataLocation()	266
6.38.3.6	getMetadata()	266
6.38.3.7	multirun_enabled()	266
6.38.3.8	output()	267
6.38.3.9	perturb()	267

6.38.3.10	reset()	267
6.38.3.11	setDataLocation()	267
6.38.3.12	verbose_print()	268
6.38.3.13	writeConfig()	268
6.38.3.14	writeConfigItem()	268
6.38.4	Member Data Documentation	269
6.38.4.1	ap_paramList	269
6.38.4.2	end_date	269
6.38.4.3	start_date	269
6.38.4.4	verbose	269
6.39	skdaccess.engineering.webcam.mit_sailing.stream.DataFetcher Class Reference	269
6.39.1	Detailed Description	270
6.39.2	Constructor & Destructor Documentation	270
6.39.2.1	__init__()	270
6.39.3	Member Function Documentation	271
6.39.3.1	__str__()	271
6.39.3.2	getConfig()	271
6.39.3.3	getConfigItem()	271
6.39.3.4	getMetadata()	272
6.39.3.5	multirun_enabled()	272
6.39.3.6	output()	272
6.39.3.7	perturb()	273
6.39.3.8	reset()	273
6.39.3.9	retrieveOnlineData()	273
6.39.3.10	verbose_print()	273
6.39.3.11	writeConfig()	274
6.39.3.12	writeConfigItem()	274
6.39.4	Member Data Documentation	274

6.39.4.1	<code>ap_paramList</code>	275
6.39.4.2	<code>camera_list</code>	275
6.39.4.3	<code>verbose</code>	275
6.40	<code>skdaccess.planetary.ode.cache.DataFetcher Class Reference</code>	275
6.40.1	Detailed Description	277
6.40.2	Constructor & Destructor Documentation	277
6.40.2.1	<code>__init__()</code>	277
6.40.3	Member Function Documentation	277
6.40.3.1	<code>__str__()</code>	278
6.40.3.2	<code>cacheData()</code>	278
6.40.3.3	<code>checkIfDataExists()</code>	278
6.40.3.4	<code>getConfig()</code>	279
6.40.3.5	<code>getConfigItem()</code>	279
6.40.3.6	<code>getDataLocation()</code>	279
6.40.3.7	<code>getHDFStorage()</code>	281
6.40.3.8	<code>getMetadata()</code>	281
6.40.3.9	<code>multirun_enabled()</code>	282
6.40.3.10	<code>output()</code>	282
6.40.3.11	<code>perturb()</code>	282
6.40.3.12	<code>reset()</code>	282
6.40.3.13	<code>setDataLocation()</code>	282
6.40.3.14	<code>verbose_print()</code>	283
6.40.3.15	<code>writeConfig()</code>	283
6.40.3.16	<code>writeConfigItem()</code>	283
6.40.4	Member Data Documentation	284
6.40.4.1	<code>ap_paramList</code>	284
6.40.4.2	<code>eastern_lon</code>	284
6.40.4.3	<code>file_name</code>	284

6.40.4.4	instrument	284
6.40.4.5	max_lat	284
6.40.4.6	max_ob_time	285
6.40.4.7	min_lat	285
6.40.4.8	min_ob_time	285
6.40.4.9	mission	285
6.40.4.10	number_product_limit	285
6.40.4.11	product_id	285
6.40.4.12	product_type	285
6.40.4.13	remove_ndv	286
6.40.4.14	result_offset_number	286
6.40.4.15	target	286
6.40.4.16	verbose	286
6.40.4.17	western_lon	286
6.41	skdaccess.astro.voyager.DataFetcher Class Reference	286
6.41.1	Detailed Description	288
6.41.2	Constructor & Destructor Documentation	288
6.41.2.1	__init__()	288
6.41.3	Member Function Documentation	288
6.41.3.1	__str__()	288
6.41.3.2	cacheData()	289
6.41.3.3	checkIfDataExists()	289
6.41.3.4	generateURL()	290
6.41.3.5	getConfig()	290
6.41.3.6	getConfigItem()	290
6.41.3.7	getDataLocation()	291
6.41.3.8	getHDFStorage()	291
6.41.3.9	getMetadata()	292

6.41.3.10	getMetadataFiles()	292
6.41.3.11	multirun_enabled()	292
6.41.3.12	output()	293
6.41.3.13	parseVoyagerData()	293
6.41.3.14	parseVoyagerMetadata()	293
6.41.3.15	perturb()	294
6.41.3.16	reset()	294
6.41.3.17	setDataLocation()	294
6.41.3.18	verbose_print()	294
6.41.3.19	writeConfig()	295
6.41.3.20	writeConfigItem()	295
6.41.4	Member Data Documentation	296
6.41.4.1	ap_paramList	296
6.41.4.2	base_url	296
6.41.4.3	field_names	296
6.41.4.4	field_widths	296
6.41.4.5	spacecraft_list	296
6.41.4.6	verbose	296
6.41.4.7	year_list	297
6.42	skdaccess.astro.tess.data.cache.DataFetcher Class Reference	297
6.42.1	Detailed Description	297
6.42.2	Constructor & Destructor Documentation	297
6.42.2.1	__init__()	297
6.42.3	Member Function Documentation	298
6.42.3.1	generateURLFromTID()	298
6.42.3.2	getTargetInformation()	298
6.42.4	Member Data Documentation	298
6.42.4.1	end_url	298

6.42.4.2	start_url	299
6.43	skdaccess.geo.imsdnhs.DataFetcher Class Reference	299
6.43.1	Detailed Description	300
6.43.2	Constructor & Destructor Documentation	300
6.43.2.1	__init__()	300
6.43.3	Member Function Documentation	301
6.43.3.1	__str__()	301
6.43.3.2	downloadFullDataset()	301
6.43.3.3	getConfig()	301
6.43.3.4	getConfigItem()	302
6.43.3.5	getDataLocation()	302
6.43.3.6	getMetadata()	302
6.43.3.7	multirun_enabled()	303
6.43.3.8	output()	303
6.43.3.9	perturb()	303
6.43.3.10	reset()	303
6.43.3.11	setDataLocation()	303
6.43.3.12	verbose_print()	304
6.43.3.13	writeConfig()	304
6.43.3.14	writeConfigItem()	304
6.43.4	Member Data Documentation	305
6.43.4.1	ap_paramList	305
6.43.4.2	coordinate_dict	305
6.43.4.3	end_date	305
6.43.4.4	start_date	305
6.43.4.5	verbose	305
6.44	skdaccess.astro.spectra.stream.DataFetcher Class Reference	306
6.44.1	Detailed Description	307

6.44.2	Constructor & Destructor Documentation	307
6.44.2.1	__init__()	307
6.44.3	Member Function Documentation	307
6.44.3.1	__str__()	307
6.44.3.2	getConfig()	308
6.44.3.3	getConfigItem()	308
6.44.3.4	getMetadata()	308
6.44.3.5	multirun_enabled()	309
6.44.3.6	output()	309
6.44.3.7	perturb()	309
6.44.3.8	reset()	309
6.44.3.9	retrieveOnlineData()	309
6.44.3.10	verbose_print()	310
6.44.3.11	writeConfig()	310
6.44.3.12	writeConfigItem()	310
6.44.4	Member Data Documentation	311
6.44.4.1	ap_paramList	311
6.44.4.2	verbose	311
6.45	skdaccess.geo.modis.stream.DataFetcher Class Reference	311
6.45.1	Detailed Description	312
6.45.2	Constructor & Destructor Documentation	313
6.45.2.1	__init__()	313
6.45.3	Member Function Documentation	313
6.45.3.1	__str__()	313
6.45.3.2	getConfig()	314
6.45.3.3	getConfigItem()	314
6.45.3.4	getMetadata()	314
6.45.3.5	multirun_enabled()	315

6.45.3.6	output()	315
6.45.3.7	perturb()	315
6.45.3.8	reset()	315
6.45.3.9	retrieveOnlineData()	315
6.45.3.10	verbose_print()	316
6.45.3.11	writeConfig()	316
6.45.3.12	writeConfigItem()	316
6.45.4	Member Data Documentation	317
6.45.4.1	ap_paramList	317
6.45.4.2	daynightboth	317
6.45.4.3	end_date	317
6.45.4.4	grid	317
6.45.4.5	grid_fill	318
6.45.4.6	modis_id	318
6.45.4.7	modis_identifier	318
6.45.4.8	modis_platform	318
6.45.4.9	start_date	318
6.45.4.10	use_long_name	318
6.45.4.11	variable_list	318
6.45.4.12	verbose	319
6.46	skdaccess.geo.grace.mascon.cache.DataFetcher Class Reference	319
6.46.1	Detailed Description	320
6.46.2	Constructor & Destructor Documentation	320
6.46.2.1	__init__()	320
6.46.3	Member Function Documentation	321
6.46.3.1	__str__()	321
6.46.3.2	cacheData()	321
6.46.3.3	checkIfDataExists()	322

6.46.3.4	getConfig()	322
6.46.3.5	getConfigItem()	322
6.46.3.6	getDataLocation()	323
6.46.3.7	getHDFSStorage()	323
6.46.3.8	getMasconPlacement()	324
6.46.3.9	getMetadata()	324
6.46.3.10	multirun_enabled()	324
6.46.3.11	output()	324
6.46.3.12	perturb()	325
6.46.3.13	reset()	325
6.46.3.14	setDataLocation()	325
6.46.3.15	verbose_print()	325
6.46.3.16	writeConfig()	326
6.46.3.17	writeConfigItem()	326
6.46.4	Member Data Documentation	326
6.46.4.1	ap_paramList	327
6.46.4.2	end_date	327
6.46.4.3	mascon_placement_url	327
6.46.4.4	mascon_url	327
6.46.4.5	scale_factor_url	327
6.46.4.6	start_date	327
6.46.4.7	verbose	327
6.47	skdaccess.geo.sentinel_1.cache.DataFetcher Class Reference	328
6.47.1	Detailed Description	329
6.47.2	Constructor & Destructor Documentation	329
6.47.2.1	__init__()	329
6.47.3	Member Function Documentation	330
6.47.3.1	__str__()	330

6.47.3.2	cacheData()	330
6.47.3.3	checkIfDataExists()	331
6.47.3.4	getConfig()	331
6.47.3.5	getConfigItem()	331
6.47.3.6	getDataLocation()	332
6.47.3.7	getHDFSStorage()	332
6.47.3.8	getMetadata()	333
6.47.3.9	multirun_enabled()	333
6.47.3.10	output()	333
6.47.3.11	perturb()	333
6.47.3.12	reset()	334
6.47.3.13	setDataLocation()	334
6.47.3.14	verbose_print()	334
6.47.3.15	writeConfig()	334
6.47.3.16	writeConfigItem()	335
6.47.4	Member Data Documentation	335
6.47.4.1	ap_paramList	335
6.47.4.2	local_paths	335
6.47.4.3	password	336
6.47.4.4	polarization	336
6.47.4.5	satellite_url_list	336
6.47.4.6	swath	336
6.47.4.7	url_list	336
6.47.4.8	username	336
6.47.4.9	verbose	336
6.48	skdaccess.framework.data_class.DataFetcherBase Class Reference	337
6.48.1	Detailed Description	338
6.48.2	Constructor & Destructor Documentation	338

6.48.2.1	<code>__init__()</code>	338
6.48.3	Member Function Documentation	338
6.48.3.1	<code>__str__()</code>	338
6.48.3.2	<code>getConfig()</code>	338
6.48.3.3	<code>getConfigItem()</code>	339
6.48.3.4	<code>getMetadata()</code>	339
6.48.3.5	<code>multirun_enabled()</code>	339
6.48.3.6	<code>output()</code>	340
6.48.3.7	<code>perturb()</code>	340
6.48.3.8	<code>reset()</code>	340
6.48.3.9	<code>verbose_print()</code>	340
6.48.3.10	<code>writeConfig()</code>	341
6.48.3.11	<code>writeConfigItem()</code>	341
6.48.4	Member Data Documentation	341
6.48.4.1	<code>ap_paramList</code>	341
6.48.4.2	<code>verbose</code>	342
6.49	<code>skdaccess.framework.data_class.DataFetcherCache</code> Class Reference	342
6.49.1	Detailed Description	343
6.49.2	Member Function Documentation	344
6.49.2.1	<code>__str__()</code>	344
6.49.2.2	<code>cacheData()</code>	344
6.49.2.3	<code>checkIfDataExists()</code>	345
6.49.2.4	<code>getConfig()</code>	345
6.49.2.5	<code>getConfigItem()</code>	345
6.49.2.6	<code>getDataLocation()</code>	346
6.49.2.7	<code>getHDFStorage()</code>	346
6.49.2.8	<code>getMetadata()</code>	346
6.49.2.9	<code>multirun_enabled()</code>	347

6.49.2.10	output()	347
6.49.2.11	perturb()	347
6.49.2.12	reset()	347
6.49.2.13	setDataLocation()	347
6.49.2.14	verbose_print()	348
6.49.2.15	writeConfig()	348
6.49.2.16	writeConfigItem()	348
6.49.3	Member Data Documentation	349
6.49.3.1	ap_paramList	349
6.49.3.2	verbose	349
6.50	skdaccess.framework.data_class.DataFetcherLocal Class Reference	349
6.50.1	Detailed Description	350
6.50.2	Member Function Documentation	350
6.50.2.1	__str__()	351
6.50.2.2	getConfig()	351
6.50.2.3	getConfigItem()	351
6.50.2.4	getDataLocation()	351
6.50.2.5	getMetadata()	352
6.50.2.6	multirun_enabled()	352
6.50.2.7	output()	352
6.50.2.8	perturb()	353
6.50.2.9	reset()	353
6.50.2.10	setDataLocation()	353
6.50.2.11	verbose_print()	353
6.50.2.12	writeConfig()	354
6.50.2.13	writeConfigItem()	354
6.50.3	Member Data Documentation	354
6.50.3.1	ap_paramList	355

6.50.3.2	verbose	355
6.51	skdaccess.framework.data_class.DataFetcherStorage Class Reference	355
6.51.1	Detailed Description	356
6.51.2	Member Function Documentation	356
6.51.2.1	__str__()	356
6.51.2.2	downloadFullDataset()	356
6.51.2.3	getConfig()	357
6.51.2.4	getConfigItem()	357
6.51.2.5	getDataLocation()	358
6.51.2.6	getMetadata()	358
6.51.2.7	multirun_enabled()	358
6.51.2.8	output()	359
6.51.2.9	perturb()	359
6.51.2.10	reset()	359
6.51.2.11	setDataLocation()	359
6.51.2.12	verbose_print()	360
6.51.2.13	writeConfig()	360
6.51.2.14	writeConfigItem()	360
6.51.3	Member Data Documentation	361
6.51.3.1	ap_paramList	361
6.51.3.2	verbose	361
6.52	skdaccess.framework.data_class.DataFetcherStream Class Reference	361
6.52.1	Detailed Description	362
6.52.2	Member Function Documentation	362
6.52.2.1	__str__()	363
6.52.2.2	getConfig()	363
6.52.2.3	getConfigItem()	363
6.52.2.4	getMetadata()	363

6.52.2.5	multirun_enabled()	364
6.52.2.6	output()	364
6.52.2.7	perturb()	364
6.52.2.8	reset()	364
6.52.2.9	retrieveOnlineData()	364
6.52.2.10	verbose_print()	365
6.52.2.11	writeConfig()	365
6.52.2.12	writeConfigItem()	365
6.52.3	Member Data Documentation	366
6.52.3.1	ap_paramList	366
6.52.3.2	verbose	366
6.53	skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper Class Reference	366
6.53.1	Detailed Description	367
6.53.2	Member Function Documentation	367
6.53.2.1	__len__()	368
6.53.2.2	addResult()	368
6.53.2.3	get()	368
6.53.2.4	getIterator()	369
6.53.2.5	getResults()	369
6.53.2.6	getRunID()	369
6.53.2.7	info()	369
6.53.2.8	reset()	370
6.53.2.9	update()	370
6.53.2.10	updateMetadata()	370
6.53.3	Member Data Documentation	370
6.53.3.1	constants	370
6.53.3.2	data	371
6.53.3.3	meta_data	371

6.53.3.4	results	371
6.53.3.5	run_id	371
6.54	skdaccess.framework.data_class.DataWrapperBase Class Reference	371
6.54.1	Detailed Description	372
6.54.2	Constructor & Destructor Documentation	372
6.54.2.1	__init__()	372
6.54.3	Member Function Documentation	373
6.54.3.1	__len__()	373
6.54.3.2	addResult()	373
6.54.3.3	get()	373
6.54.3.4	getIterator()	374
6.54.3.5	getResults()	374
6.54.3.6	getRunID()	374
6.54.3.7	info()	375
6.54.3.8	reset()	375
6.54.3.9	update()	375
6.54.3.10	updateMetadata()	375
6.54.4	Member Data Documentation	376
6.54.4.1	constants	376
6.54.4.2	data	376
6.54.4.3	meta_data	376
6.54.4.4	results	376
6.54.4.5	run_id	376
6.55	skdaccess.utilities.file_browser.FileBrowser Class Reference	377
6.55.1	Constructor & Destructor Documentation	377
6.55.1.1	__init__()	377
6.55.2	Member Function Documentation	377
6.55.2.1	widget()	377

6.55.3	Member Data Documentation	378
6.55.3.1	dirs	378
6.55.3.2	files	378
6.55.3.3	path	378
6.56	skdaccess.framework.data_class.ImageWrapper Class Reference	378
6.56.1	Detailed Description	379
6.56.2	Member Function Documentation	379
6.56.2.1	__len__()	380
6.56.2.2	addResult()	380
6.56.2.3	deleteData()	380
6.56.2.4	get()	380
6.56.2.5	getIterator()	381
6.56.2.6	getResults()	381
6.56.2.7	getRunID()	381
6.56.2.8	info()	382
6.56.2.9	reset()	382
6.56.2.10	update()	382
6.56.2.11	updateData()	382
6.56.2.12	updateMetadata()	383
6.56.3	Member Data Documentation	383
6.56.3.1	constants	383
6.56.3.2	data	383
6.56.3.3	meta_data	383
6.56.3.4	results	384
6.56.3.5	run_id	384
6.57	skdaccess.utilities.modis_util.LatLon Class Reference	384
6.57.1	Detailed Description	385
6.57.2	Constructor & Destructor Documentation	385

6.57.2.1	__init__()	385
6.57.3	Member Function Documentation	385
6.57.3.1	__call__()	385
6.57.4	Member Data Documentation	386
6.57.4.1	alat	386
6.57.4.2	alon	386
6.57.4.3	lat_data	386
6.57.4.4	lon_data	386
6.57.4.5	x_offset	386
6.57.4.6	y_offset	386
6.58	skdaccess.utilities.image_util.LinearGeolocation Class Reference	387
6.58.1	Detailed Description	387
6.58.2	Constructor & Destructor Documentation	388
6.58.2.1	__init__()	388
6.58.3	Member Function Documentation	388
6.58.3.1	getExtents()	388
6.58.3.2	getLatLon()	388
6.58.3.3	getYX()	389
6.58.4	Member Data Documentation	389
6.58.4.1	flip_y	389
6.58.4.2	lat_extents	389
6.58.4.3	lat_pixel_size	390
6.58.4.4	len_x	390
6.58.4.5	len_y	390
6.58.4.6	lon_extents	390
6.58.4.7	lon_pixel_size	390
6.58.4.8	start_lat	390
6.58.4.9	start_lon	390

6.58.4.10	x_offset	391
6.58.4.11	y_offset	391
6.59	skdaccess.framework.data_class.SeriesDictionaryWrapper Class Reference	391
6.59.1	Detailed Description	392
6.59.2	Member Function Documentation	392
6.59.2.1	__len__()	392
6.59.2.2	addResult()	392
6.59.2.3	get()	393
6.59.2.4	getIndices()	393
6.59.2.5	getIterator()	393
6.59.2.6	getLength()	394
6.59.2.7	getResults()	394
6.59.2.8	getRunID()	394
6.59.2.9	info()	394
6.59.2.10	reset()	395
6.59.2.11	update()	395
6.59.2.12	updateMetadata()	395
6.59.3	Member Data Documentation	395
6.59.3.1	constants	395
6.59.3.2	data	396
6.59.3.3	data_names	396
6.59.3.4	error_names	396
6.59.3.5	meta_data	396
6.59.3.6	results	396
6.59.3.7	run_id	396
6.60	skdaccess.framework.data_class.SeriesWrapper Class Reference	397
6.60.1	Detailed Description	398
6.60.2	Constructor & Destructor Documentation	398

6.60.2.1	<code>__init__()</code>	398
6.60.3	Member Function Documentation	398
6.60.3.1	<code>__len__()</code>	399
6.60.3.2	<code>addResult()</code>	399
6.60.3.3	<code>get()</code>	399
6.60.3.4	<code>getIndices()</code>	400
6.60.3.5	<code>getIterator()</code>	400
6.60.3.6	<code>getLength()</code>	400
6.60.3.7	<code>getResults()</code>	400
6.60.3.8	<code>getRunID()</code>	401
6.60.3.9	<code>info()</code>	401
6.60.3.10	<code>reset()</code>	401
6.60.3.11	<code>update()</code>	401
6.60.3.12	<code>updateMetadata()</code>	402
6.60.4	Member Data Documentation	402
6.60.4.1	<code>constants</code>	402
6.60.4.2	<code>data</code>	402
6.60.4.3	<code>data_names</code>	402
6.60.4.4	<code>error_names</code>	403
6.60.4.5	<code>meta_data</code>	403
6.60.4.6	<code>results</code>	403
6.60.4.7	<code>run_id</code>	403
6.61	<code>skdaccess.utilities.sounding_util.SoundingParser</code> Class Reference	403
6.61.1	Detailed Description	404
6.61.2	Constructor & Destructor Documentation	404
6.61.2.1	<code>__init__()</code>	404
6.61.3	Member Function Documentation	404
6.61.3.1	<code>handle_data()</code>	404

6.61.3.2	handle_endtag()	405
6.61.3.3	handle_starttag()	405
6.61.4	Member Data Documentation	405
6.61.4.1	data_dict	405
6.61.4.2	in_header	406
6.61.4.3	in_pre_tag	406
6.61.4.4	label	406
6.61.4.5	metadata_dict	406
6.61.4.6	read_data	406
6.61.4.7	tmp	406
6.62	skdaccess.utilities.image_util.SplineLatLon Class Reference	407
6.62.1	Detailed Description	407
6.62.2	Constructor & Destructor Documentation	407
6.62.2.1	__init__()	408
6.62.3	Member Function Documentation	408
6.62.3.1	__call__()	408
6.62.4	Member Data Documentation	409
6.62.4.1	lat_func	409
6.62.4.2	lon_func	409
6.62.4.3	x_offset	409
6.62.4.4	y_offset	409
6.63	skdaccess.framework.data_class.TableWrapper Class Reference	410
6.63.1	Detailed Description	411
6.63.2	Constructor & Destructor Documentation	411
6.63.2.1	__init__()	411
6.63.3	Member Function Documentation	412
6.63.3.1	__len__()	412
6.63.3.2	addColumn()	412

6.63.3.3	addResult()	412
6.63.3.4	get()	413
6.63.3.5	getDefaultColumns()	413
6.63.3.6	getDefaultErrorColumns()	413
6.63.3.7	getIterator()	414
6.63.3.8	getLength()	414
6.63.3.9	getResults()	414
6.63.3.10	getRunID()	414
6.63.3.11	info()	415
6.63.3.12	removeFrames()	415
6.63.3.13	reset()	415
6.63.3.14	update()	415
6.63.3.15	updateData()	416
6.63.3.16	updateFrames()	416
6.63.3.17	updateMetadata()	416
6.63.4	Member Data Documentation	417
6.63.4.1	constants	417
6.63.4.2	data	417
6.63.4.3	default_columns	417
6.63.4.4	default_error_columns	417
6.63.4.5	meta_data	417
6.63.4.6	results	418
6.63.4.7	run_id	418
6.64	skdaccess.framework.data_class.XArrayWrapper Class Reference	418
6.64.1	Detailed Description	419
6.64.2	Constructor & Destructor Documentation	419
6.64.2.1	__init__()	419
6.64.3	Member Function Documentation	419

6.64.3.1	__len__()	419
6.64.3.2	addResult()	419
6.64.3.3	get()	420
6.64.3.4	getIterator()	420
6.64.3.5	getResults()	420
6.64.3.6	getRunID()	421
6.64.3.7	info()	421
6.64.3.8	reset()	421
6.64.3.9	update()	421
6.64.3.10	updateMetadata()	422
6.64.4	Member Data Documentation	422
6.64.4.1	constants	422
6.64.4.2	data	422
6.64.4.3	index_list	422
6.64.4.4	meta_data	423
6.64.4.5	results	423
6.64.4.6	run_id	423
7	File Documentation	425
7.1	astro/spectra/stream.py File Reference	425
7.2	finance/timeseries/stream.py File Reference	425
7.3	engineering/webcam/mit_sailing/stream.py File Reference	425
7.4	engineering/la/traffic_counts/stream.py File Reference	426
7.5	engineering/la/generic/stream.py File Reference	426
7.6	astro/tess/generic/cache.py File Reference	426
7.7	astro/tess/data/cache.py File Reference	427
7.8	astro/tess/simulated/cache.py File Reference	427
7.9	examples/terminal_groundwater_example.py File Reference	427

7.10 framework/data_class.py File Reference	428
7.11 framework/param_class.py File Reference	428
7.12 geo/era_interim/cache/data_fetcher.py File Reference	429
7.13 geo/modis/cache/reflectance/data_fetcher.py File Reference	429
7.14 geo/modis/cache/data_fetcher.py File Reference	429
7.15 geo/modis/cache/cloud_opacity/data_fetcher.py File Reference	430
7.16 geo/modis/cache/cloud_mask/data_fetcher.py File Reference	430
7.17 geo/modis/stream/reflectance/data_fetcher.py File Reference	430
7.18 geo/modis/stream/data_fetcher.py File Reference	431
7.19 geo/modis/stream/cloud_opacity/data_fetcher.py File Reference	431
7.20 geo/modis/stream/cloud_mask/data_fetcher.py File Reference	431
7.21 geo/gldas/data_fetcher.py File Reference	431
7.22 geo/uavsar/cache/data_fetcher.py File Reference	432
7.23 geo/pbo/data_fetcher.py File Reference	432
7.24 geo/grace/data_fetcher.py File Reference	432
7.25 geo/grace/mascon/cache/data_fetcher.py File Reference	433
7.26 geo/sentinel_1/cache/data_fetcher.py File Reference	433
7.27 geo/groundwater/data_fetcher.py File Reference	433
7.28 geo/srtm/cache/data_fetcher.py File Reference	433
7.29 geo/mahali/temperature/data_fetcher.py File Reference	434
7.30 geo/mahali/rinex/data_fetcher.py File Reference	434
7.31 geo/mahali/tec/data_fetcher.py File Reference	434
7.32 geo/magnetometer/data_fetcher.py File Reference	435
7.33 geo/ngl_gps/data_fetcher.py File Reference	435
7.34 geo/wyoming_sounding/cache/data_fetcher.py File Reference	435
7.35 geo/wyoming_sounding/stream/data_fetcher.py File Reference	435
7.36 geo/imsdnhs/data_fetcher.py File Reference	436
7.37 astro/kepler/data_fetcher.py File Reference	436

7.38 astro/voyager/data_fetcher.py File Reference	436
7.39 solar/sdo/data_fetcher.py File Reference	437
7.40 planetary/ode/cache/data_fetcher.py File Reference	437
7.41 geo/mahali/rinex/data_wrapper.py File Reference	437
7.42 utilities/file_browser.py File Reference	437
7.43 utilities/file_util.py File Reference	438
7.44 utilities/grace_util.py File Reference	438
7.45 utilities/gw_util.py File Reference	438
7.46 utilities/image_util.py File Reference	439
7.47 utilities/kepler_util.py File Reference	439
7.48 utilities/mahali_util.py File Reference	440
7.49 utilities/modis_util.py File Reference	440
7.50 utilities/ode_util.py File Reference	441
7.51 utilities/pbo_util.py File Reference	441
7.52 utilities/sentinel_1_util.py File Reference	442
7.53 utilities/sounding_util.py File Reference	442
7.54 utilities/srtm_util.py File Reference	442
7.55 utilities/support.py File Reference	443
7.56 utilities/tess_utils.py File Reference	443
7.57 utilities/uavsar_util.py File Reference	443

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

skdaccess	13
skdaccess.astro	13
skdaccess.astro.kepler	13
skdaccess.astro.kepler.data_fetcher	14
skdaccess.astro.spectra	14
skdaccess.astro.spectra.stream	14
skdaccess.astro.tess	14
skdaccess.astro.tess.data	14
skdaccess.astro.tess.data.cache	14
skdaccess.astro.tess.generic	15
skdaccess.astro.tess.generic.cache	15
skdaccess.astro.tess.simulated	15
skdaccess.astro.tess.simulated.cache	15
skdaccess.astro.voyager	15
skdaccess.astro.voyager.data_fetcher	15
skdaccess.engineering	16
skdaccess.engineering.la	16
skdaccess.engineering.la.generic	16
skdaccess.engineering.la.generic.stream	16
skdaccess.engineering.la.traffic_counts	16
skdaccess.engineering.la.traffic_counts.stream	16
skdaccess.engineering.webcam	17
skdaccess.engineering.webcam.mit_sailing	17
skdaccess.engineering.webcam.mit_sailing.stream	17
skdaccess.finance	17
skdaccess.finance.timeseries	17
skdaccess.finance.timeseries.stream	17
skdaccess.framework	18
skdaccess.framework.data_class	18
skdaccess.framework.param_class	18
skdaccess.geo	19

skdaccess.geo.era_interim	19
skdaccess.geo.era_interim.cache	19
skdaccess.geo.era_interim.cache.data_fetcher	20
skdaccess.geo.gldas	20
skdaccess.geo.gldas.data_fetcher	20
skdaccess.geo.grace	20
skdaccess.geo.grace.data_fetcher	20
skdaccess.geo.grace.mascon	20
skdaccess.geo.grace.mascon.cache	21
skdaccess.geo.grace.mascon.cache.data_fetcher	21
skdaccess.geo.groundwater	21
skdaccess.geo.groundwater.data_fetcher	21
skdaccess.geo.imsdnh	21
skdaccess.geo.imsdnh.data_fetcher	21
skdaccess.geo.magnetometer	22
skdaccess.geo.magnetometer.data_fetcher	22
skdaccess.geo.mahali	22
skdaccess.geo.mahali.rinex	22
skdaccess.geo.mahali.rinex.data_fetcher	22
skdaccess.geo.mahali.rinex.data_wrapper	22
skdaccess.geo.mahali.tec	23
skdaccess.geo.mahali.tec.data_fetcher	23
skdaccess.geo.mahali.temperature	23
skdaccess.geo.mahali.temperature.data_fetcher	23
skdaccess.geo.modis	23
skdaccess.geo.modis.cache	23
skdaccess.geo.modis.cache.cloud_mask	24
skdaccess.geo.modis.cache.cloud_mask.data_fetcher	24
skdaccess.geo.modis.cache.cloud_opacity	24
skdaccess.geo.modis.cache.cloud_opacity.data_fetcher	24
skdaccess.geo.modis.cache.data_fetcher	24
skdaccess.geo.modis.cache.reflectance	24
skdaccess.geo.modis.cache.reflectance.data_fetcher	25
skdaccess.geo.modis.stream	25
skdaccess.geo.modis.stream.cloud_mask	25
skdaccess.geo.modis.stream.cloud_mask.data_fetcher	25
skdaccess.geo.modis.stream.cloud_opacity	25
skdaccess.geo.modis.stream.cloud_opacity.data_fetcher	25
skdaccess.geo.modis.stream.data_fetcher	26
skdaccess.geo.modis.stream.reflectance	26
skdaccess.geo.modis.stream.reflectance.data_fetcher	26
skdaccess.geo.ngl_gps	26
skdaccess.geo.ngl_gps.data_fetcher	26
skdaccess.geo.pbo	26
skdaccess.geo.pbo.data_fetcher	27
skdaccess.geo.sentinel_1	27
skdaccess.geo.sentinel_1.cache	27
skdaccess.geo.sentinel_1.cache.data_fetcher	27
skdaccess.geo.srtm	27
skdaccess.geo.srtm.cache	27
skdaccess.geo.srtm.cache.data_fetcher	28
skdaccess.geo.uavsar	28
skdaccess.geo.uavsar.cache	28
skdaccess.geo.uavsar.cache.data_fetcher	28

skdaccess.geo.wyoming_sounding	28
skdaccess.geo.wyoming_sounding.cache	28
skdaccess.geo.wyoming_sounding.cache.data_fetcher	29
skdaccess.geo.wyoming_sounding.stream	29
skdaccess.geo.wyoming_sounding.stream.data_fetcher	29
skdaccess.planetary	29
skdaccess.planetary.ode	29
skdaccess.planetary.ode.cache	29
skdaccess.planetary.ode.cache.data_fetcher	30
skdaccess.solar	30
skdaccess.solar.sdo	30
skdaccess.solar.sdo.data_fetcher	30
skdaccess.utilities	30
skdaccess.utilities.file_browser	31
skdaccess.utilities.file_util	31
skdaccess.utilities.grace_util	31
skdaccess.utilities.gw_util	34
skdaccess.utilities.image_util	36
skdaccess.utilities.kepler_util	39
skdaccess.utilities.mahali_util	39
skdaccess.utilities.modis_util	40
skdaccess.utilities.ode_util	45
skdaccess.utilities.pbo_util	49
skdaccess.utilities.sentinel_1_util	53
skdaccess.utilities.sounding_util	54
skdaccess.utilities.srtm_util	55
skdaccess.utilities.support	56
skdaccess.utilities.tess_utils	58
skdaccess.utilities.uavsar_util	59
terminal_groundwater_example	59

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

skdaccess.framework.param_class.AutoParam	86
skdaccess.framework.param_class.AutoParamList	89
skdaccess.framework.param_class.AutoParamListCycle	92
skdaccess.framework.param_class.AutoParamMinMax	95
GenericDataFetcher	
skdaccess.engineering.la.traffic_counts.stream.DataFetcher	199
GenericDF	
skdaccess.astro.tess.data.cache.DataFetcher	297
skdaccess.astro.tess.simulated.cache.DataFetcher	113
MDF	
skdaccess.geo.modis.cache.cloud_mask.DataFetcher	182
skdaccess.geo.modis.cache.cloud_opacity.DataFetcher	173
skdaccess.geo.modis.cache.reflectance.DataFetcher	165
skdaccess.geo.modis.stream.cloud_mask.DataFetcher	174
skdaccess.geo.modis.stream.cloud_opacity.DataFetcher	115
skdaccess.geo.modis.stream.reflectance.DataFetcher	242
object	
skdaccess.framework.data_class.DataFetcherBase	337
skdaccess.framework.data_class.DataFetcherLocal	349
skdaccess.framework.data_class.DataFetcherCache	342
skdaccess.astro.kepler.DataFetcher	133
skdaccess.astro.tess.generic.cache.DataFetcher	116
skdaccess.astro.voyager.DataFetcher	286
skdaccess.geo.era_interim.cache.DataFetcher	104
skdaccess.geo.grace.mascon.cache.DataFetcher	319
skdaccess.geo.mahali.rinex.DataFetcher	183
skdaccess.geo.mahali.tec.DataFetcher	200
skdaccess.geo.modis.cache.DataFetcher	231
skdaccess.geo.sentinel_1.cache.DataFetcher	328
skdaccess.geo.srtm.cache.DataFetcher	156
skdaccess.geo.uavsar.cache.DataFetcher	215

skdaccess.geo.wyoming_sounding.cache.DataFetcher	243
skdaccess.planetary.ode.cache.DataFetcher	275
skdaccess.framework.data_class.DataFetcherStorage	355
skdaccess.geo.gldas.DataFetcher	192
skdaccess.geo.grace.DataFetcher	262
skdaccess.geo.groundwater.DataFetcher	125
skdaccess.geo.imsdnhs.DataFetcher	299
skdaccess.geo.ngl_gps.DataFetcher	223
skdaccess.geo.pbo.DataFetcher	252
skdaccess.framework.data_class.DataFetcherStream	361
skdaccess.astro.spectra.stream.DataFetcher	306
skdaccess.engineering.la.generic.stream.DataFetcher	141
skdaccess.engineering.webcam.mit_sailing.stream.DataFetcher	269
skdaccess.finance.timeseries.stream.DataFetcher	175
skdaccess.geo.magnetometer.DataFetcher	208
skdaccess.geo.mahali.temperature.DataFetcher	166
skdaccess.geo.modis.stream.DataFetcher	311
skdaccess.geo.wyoming_sounding.stream.DataFetcher	148
skdaccess.solar.sdo.DataFetcher	98
skdaccess.framework.data_class.DataWrapperBase	371
skdaccess.framework.data_class.ImageWrapper	378
skdaccess.framework.data_class.SeriesWrapper	397
skdaccess.framework.data_class.SeriesDictionaryWrapper	391
skdaccess.framework.data_class.TableWrapper	410
skdaccess.framework.data_class.XArrayWrapper	418
skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper	366
skdaccess.framework.param_class.AutoList	65
skdaccess.framework.param_class.AutoListCycle	69
skdaccess.framework.param_class.AutoListPermute	74
skdaccess.framework.param_class.AutoListRemove	78
skdaccess.framework.param_class.AutoListSubset	82
skdaccess.utilities.file_browser.FileBrowser	377
skdaccess.utilities.image_util.AffineGlobalCoords	63
skdaccess.utilities.image_util.LinearGeolocation	387
skdaccess.utilities.image_util.SplineLatLon	407
skdaccess.utilities.modis_util.LatLon	384
HTMLParser	
skdaccess.utilities.sounding_util.SoundingParser	403

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

skdaccess.utilities.image_util.AffineGlobalCoords	
Convert between projected and pixel coordinates using an affine transformation	63
skdaccess.framework.param_class.AutoList	
Specifies a list for returning selections of lists, as opposed to a single element	65
skdaccess.framework.param_class.AutoListCycle	
An Autolist that cycles through different lists	69
skdaccess.framework.param_class.AutoListPermute	
A perturber that permutes a list	74
skdaccess.framework.param_class.AutoListRemove	
Removes a different single element from the initial list at each perturb call	78
skdaccess.framework.param_class.AutoListSubset	
An AutoList perturber that creates random subsets of a list	82
skdaccess.framework.param_class.AutoParam	
Defines a tunable parameter class inherited by specific subclasses	86
skdaccess.framework.param_class.AutoParamList	
A tunable parameter with a specified list of choices that can be randomly selected via perturb	89
skdaccess.framework.param_class.AutoParamListCycle	
Cycles through a list of paramters	92
skdaccess.framework.param_class.AutoParamMinMax	
A tunable parameter with min and max ranges, perturbs to a random value in range	95
skdaccess.solar.sdo.DataFetcher	
Data Fetcher for the Solar Dynamics Observatory	98
skdaccess.geo.era_interim.cache.DataFetcher	
DataFetcher for retrieving ERA-I data	104
skdaccess.astro.tess.simulated.cache.DataFetcher	
Data Fetcher for TESS data alerts	113
skdaccess.geo.modis.stream.cloud_opacity.DataFetcher	
Data Fetcher for MODIS Cloud Opacity	115
skdaccess.astro.tess.generic.cache.DataFetcher	
Data Fetcher for TESS data alerts	116
skdaccess.geo.groundwater.DataFetcher	
Generates Data Wrappers of groundwater measurements taken in the US	125

skdaccess.astro.kepler.DataFetcher	
Data Fetcher for Kepler light curve data	133
skdaccess.engineering.la.generic.stream.DataFetcher	
Class for handling data requests to data.lacity.org	141
skdaccess.geo.wyoming_sounding.stream.DataFetcher	
DataFetcher for retrieving Wyoming Sounding data	148
skdaccess.geo.srtm.cache.DataFetcher	
DataFetcher for retrieving data from the Shuttle Radar Topography Mission	156
skdaccess.geo.modis.cache.reflectance.DataFetcher	
Data fetcher for the modis surface reflectance product ('09', 1 km resolution)	165
skdaccess.geo.mahali.temperature.DataFetcher	
Data Fetcher for Mahali temperature data	166
skdaccess.geo.modis.cache.cloud_opacity.DataFetcher	
Data Fetcher for MODIS Cloud Opacity	173
skdaccess.geo.modis.stream.cloud_mask.DataFetcher	
Data Fetcher for MODIS Cloud Mask	174
skdaccess.finance.timeseries.stream.DataFetcher	
Data Fetcher for retrieving stock data	175
skdaccess.geo.modis.cache.cloud_mask.DataFetcher	
Data Fetcher for MODIS Cloud Mask	182
skdaccess.geo.mahali.rinex.DataFetcher	
Data Fetcher for Mahali Data	183
skdaccess.geo.gldas.DataFetcher	
Data Fetcher for GLDAS data	192
skdaccess.engineering.la.traffic_counts.stream.DataFetcher	
DataFetcher for retrieving traffic counts from LA	199
skdaccess.geo.mahali.tec.DataFetcher	
Data Fetcher for Mahali Data	200
skdaccess.geo.magnetometer.DataFetcher	
Data fetcher for USGS geomagnetic observatories	208
skdaccess.geo.uavsar.cache.DataFetcher	
Data Fetcher for UAVSAR data	215
skdaccess.geo.ngl_gps.DataFetcher	
Data fetcher for GPS data from Nevada Geodetic Laboratory	223
skdaccess.geo.modis.cache.DataFetcher	
Data Fetcher for MODIS data	231
skdaccess.geo.modis.stream.reflectance.DataFetcher	
Data fetcher for the modis surface reflectance product ('09', 1 km resolution)	242
skdaccess.geo.wyoming_sounding.cache.DataFetcher	
DataFetcher for retrieving Wyoming Sounding data	243
skdaccess.geo.pbo.DataFetcher	
Data fetcher for PBO GPS data	252
skdaccess.geo.grace.DataFetcher	
Data Fetcher for GRACE data	262
skdaccess.engineering.webcam.mit_sailing.stream.DataFetcher	
Data Fetcher for retrieving webcam images from the MIT Sailing Pavilion	269
skdaccess.planetary.ode.cache.DataFetcher	
Data Fetcher from the Orbital Data Explorer (ODE)	275
skdaccess.astro.voyager.DataFetcher	
Data Fetcher for Mahali temperature data	286
skdaccess.astro.tess.data.cache.DataFetcher	
Data Fetcher for TESS data alerts	297

skdaccess.geo.imsdnhs.DataFetcher	
Fetches data for the Interactive Multisensor Snow and Ice Mapping System Daily Northern Hemisphere Snow and Ice Analysis	299
skdaccess.astro.spectra.stream.DataFetcher	
Data Fetcher for Sloan Digital Sky Survey spectra	306
skdaccess.geo.modis.stream.DataFetcher	
Data Fetcher for MODIS data	311
skdaccess.geo.grace.mascon.cache.DataFetcher	
Data Fetcher for GRACE mascon data	319
skdaccess.geo.sentinel_1.cache.DataFetcher	
DataFetcher for retrieving Sentinel SLC data	328
skdaccess.framework.data_class.DataFetcherBase	
Base class for all data fetchers	337
skdaccess.framework.data_class.DataFetcherCache	
Data fetcher base class for downloading data and caching results on hard disk	342
skdaccess.framework.data_class.DataFetcherLocal	
Data fetcher base class for use when storing data locally	349
skdaccess.framework.data_class.DataFetcherStorage	
Data fetcher base class for use when entire data set is downloaded	355
skdaccess.framework.data_class.DataFetcherStream	
Data fetcher base class for downloading data into memory	361
skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper	
Data wrapper for Mahali data	366
skdaccess.framework.data_class.DataWrapperBase	
Base class for wrapping data for use in DiscoveryPipeline	371
skdaccess.utilities.file_browser.FileBrowser	
.	377
skdaccess.framework.data_class.ImageWrapper	
Wrapper for image data	378
skdaccess.utilities.modis_util.LatLon	
Calculates Lat/Lon position from y,x pixel coordinate	384
skdaccess.utilities.image_util.LinearGeolocation	
This class provides functions to convert between pixel and geodetic coordinates	387
skdaccess.framework.data_class.SeriesDictionaryWrapper	
Data wrapper for series data using a dictionary of data frames	391
skdaccess.framework.data_class.SeriesWrapper	
Data wrapper for series data using a data panel	397
skdaccess.utilities.sounding_util.SoundingParser	
This class parses Wyoming Sounding data	403
skdaccess.utilities.image_util.SplineLatLon	
Holds a 2d spline for interpolating lat/lon grid	407
skdaccess.framework.data_class.TableWrapper	
Data wrapper for table data using an ordered dictionary	410
skdaccess.framework.data_class.XArrayWrapper	
Wrapper for xarrays	418

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

astro/kepler/ data_fetcher.py	436
astro/spectra/ stream.py	425
astro/tess/data/ cache.py	427
astro/tess/generic/ cache.py	426
astro/tess/simulated/ cache.py	427
astro/voyager/ data_fetcher.py	436
engineering/la/generic/ stream.py	426
engineering/la/traffic_counts/ stream.py	426
engineering/webcam/mit_sailing/ stream.py	425
examples/ terminal_groundwater_example.py	427
finance/timeseries/ stream.py	425
framework/ data_class.py	428
framework/ param_class.py	428
geo/era_interim/cache/ data_fetcher.py	429
geo/gldas/ data_fetcher.py	431
geo/grace/ data_fetcher.py	432
geo/grace/mascon/cache/ data_fetcher.py	433
geo/groundwater/ data_fetcher.py	433
geo/imsdnhs/ data_fetcher.py	436
geo/magnetometer/ data_fetcher.py	435
geo/mahali/rinex/ data_fetcher.py	434
geo/mahali/rinex/ data_wrapper.py	437
geo/mahali/tec/ data_fetcher.py	434
geo/mahali/temperature/ data_fetcher.py	434
geo/modis/cache/ data_fetcher.py	429
geo/modis/cache/cloud_mask/ data_fetcher.py	430
geo/modis/cache/cloud_opacity/ data_fetcher.py	430
geo/modis/cache/reflectance/ data_fetcher.py	429
geo/modis/stream/ data_fetcher.py	431
geo/modis/stream/cloud_mask/ data_fetcher.py	431
geo/modis/stream/cloud_opacity/ data_fetcher.py	431

geo/modis/stream/reflectance/data_fetcher.py	430
geo/ngl_gps/data_fetcher.py	435
geo/pbo/data_fetcher.py	432
geo/sentinel_1/cache/data_fetcher.py	433
geo/srtm/cache/data_fetcher.py	433
geo/uavsar/cache/data_fetcher.py	432
geo/wyoming_sounding/cache/data_fetcher.py	435
geo/wyoming_sounding/stream/data_fetcher.py	435
planetary/ode/cache/data_fetcher.py	437
solar/sdo/data_fetcher.py	437
utilities/file_browser.py	437
utilities/file_util.py	438
utilities/grace_util.py	438
utilities/gw_util.py	438
utilities/image_util.py	439
utilities/kepler_util.py	439
utilities/mahali_util.py	440
utilities/modis_util.py	440
utilities/ode_util.py	441
utilities/pbo_util.py	441
utilities/sentinel_1_util.py	442
utilities/sounding_util.py	442
utilities/srtm_util.py	442
utilities/support.py	443
utilities/tess_utils.py	443
utilities/uavsar_util.py	443

Chapter 5

Namespace Documentation

5.1 skdaccess Namespace Reference

Namespaces

- [astro](#)
- [engineering](#)
- [finance](#)
- [framework](#)
- [geo](#)
- [planetary](#)
- [solar](#)
- [utilities](#)

5.2 skdaccess.astro Namespace Reference

Namespaces

- [kepler](#)
- [spectra](#)
- [tess](#)
- [voyager](#)

5.3 skdaccess.astro.kepler Namespace Reference

Namespaces

- [data_fetcher](#)

5.4 skdaccess.astro.kepler.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for Kepler light curve data.

5.5 skdaccess.astro.spectra Namespace Reference

Namespaces

- [stream](#)

5.6 skdaccess.astro.spectra.stream Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for Sloan Digital Sky Survey spectra.

5.7 skdaccess.astro.tess Namespace Reference

Namespaces

- [data](#)
- [generic](#)
- [simulated](#)

5.8 skdaccess.astro.tess.data Namespace Reference

Namespaces

- [cache](#)

5.9 skdaccess.astro.tess.data.cache Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for TESS data alerts.

5.10 skdaccess.astro.tess.generic Namespace Reference

Namespaces

- [cache](#)

5.11 skdaccess.astro.tess.generic.cache Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for TESS data alerts.

5.12 skdaccess.astro.tess.simulated Namespace Reference

Namespaces

- [cache](#)

5.13 skdaccess.astro.tess.simulated.cache Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for TESS data alerts.

5.14 skdaccess.astro.voyager Namespace Reference

Namespaces

- [data_fetcher](#)

5.15 skdaccess.astro.voyager.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for Mahali temperature data.

5.16 skdaccess.engineering Namespace Reference

Namespaces

- [la](#)
- [webcam](#)

5.17 skdaccess.engineering.la Namespace Reference

Namespaces

- [generic](#)
- [traffic_counts](#)

5.18 skdaccess.engineering.la.generic Namespace Reference

Namespaces

- [stream](#)

5.19 skdaccess.engineering.la.generic.stream Namespace Reference

Classes

- class [DataFetcher](#)
Class for handling data requests to data.lacity.org.

5.20 skdaccess.engineering.la.traffic_counts Namespace Reference

Namespaces

- [stream](#)

5.21 skdaccess.engineering.la.traffic_counts.stream Namespace Reference

Classes

- class [DataFetcher](#)
[DataFetcher](#) for retrieving traffic counts from LA.

5.22 skdaccess.engineering.webcam Namespace Reference

Namespaces

- [mit_sailing](#)

5.23 skdaccess.engineering.webcam.mit_sailing Namespace Reference

Namespaces

- [stream](#)

5.24 skdaccess.engineering.webcam.mit_sailing.stream Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for retrieving webcam images from the MIT Sailing Pavilion.

5.25 skdaccess.finance Namespace Reference

Namespaces

- [timeseries](#)

5.26 skdaccess.finance.timeseries Namespace Reference

Namespaces

- [stream](#)

5.27 skdaccess.finance.timeseries.stream Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for retrieving stock data.

5.28 skdaccess.framework Namespace Reference

Namespaces

- [data_class](#)
- [param_class](#)

5.29 skdaccess.framework.data_class Namespace Reference

Classes

- class [DataFetcherBase](#)
Base class for all data fetchers.
- class [DataFetcherCache](#)
Data fetcher base class for downloading data and caching results on hard disk.
- class [DataFetcherLocal](#)
Data fetcher base class for use when storing data locally.
- class [DataFetcherStorage](#)
Data fetcher base class for use when entire data set is downloaded.
- class [DataFetcherStream](#)
Data fetcher base class for downloading data into memory.
- class [DataWrapperBase](#)
Base class for wrapping data for use in DiscoveryPipeline.
- class [ImageWrapper](#)
Wrapper for image data.
- class [SeriesDictionaryWrapper](#)
Data wrapper for series data using a dictionary of data frames.
- class [SeriesWrapper](#)
Data wrapper for series data using a data panel.
- class [TableWrapper](#)
Data wrapper for table data using an ordered dictionary.
- class [XArrayWrapper](#)
Wrapper for xarrays.

5.30 skdaccess.framework.param_class Namespace Reference

Classes

- class [AutoList](#)
Specifies a list for returning selections of lists, as opposed to a single element.
- class [AutoListCycle](#)
An Autolist that cycles through different lists.
- class [AutoListPermute](#)

A perturber that permutes a list.

- class [AutoListRemove](#)

Removes a different single element from the initial list at each perturb call.

- class [AutoListSubset](#)

An [AutoList](#) perturber that creates random subsets of a list.

- class [AutoParam](#)

Defines a tunable parameter class inherited by specific subclasses.

- class [AutoParamList](#)

A tunable parameter with a specified list of choices that can be randomly selected via perturb.

- class [AutoParamListCycle](#)

Cycles through a list of paramters.

- class [AutoParamMinMax](#)

A tunable parameter with min and max ranges, perturbs to a random value in range.

5.31 skdaccess.geo Namespace Reference

Namespaces

- [era_interim](#)
- [gldas](#)
- [grace](#)
- [groundwater](#)
- [imsdnhs](#)
- [magnetometer](#)
- [mahali](#)
- [modis](#)
- [ngl_gps](#)
- [pbo](#)
- [sentinel_1](#)
- [srtm](#)
- [uavsar](#)
- [wyoming_sounding](#)

5.32 skdaccess.geo.era_interim Namespace Reference

Namespaces

- [cache](#)

5.33 skdaccess.geo.era_interim.cache Namespace Reference

Namespaces

- [data_fetcher](#)

5.34 skdaccess.geo.era_interim.cache.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
DataFetcher for retrieving ERA-I data.

5.35 skdaccess.geo.gldas Namespace Reference

Namespaces

- [data_fetcher](#)

5.36 skdaccess.geo.gldas.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for GLDAS data.

5.37 skdaccess.geo.grace Namespace Reference

Namespaces

- [data_fetcher](#)
- [mascon](#)

5.38 skdaccess.geo.grace.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for GRACE data.

5.39 skdaccess.geo.grace.mascon Namespace Reference

Namespaces

- [cache](#)

5.40 skdaccess.geo.grace.mascon.cache Namespace Reference

Namespaces

- [data_fetcher](#)

5.41 skdaccess.geo.grace.mascon.cache.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for GRACE mascon data.

5.42 skdaccess.geo.groundwater Namespace Reference

Namespaces

- [data_fetcher](#)

5.43 skdaccess.geo.groundwater.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Generates Data Wrappers of groundwater measurements taken in the US.

5.44 skdaccess.geo.imsdnhs Namespace Reference

Namespaces

- [data_fetcher](#)

5.45 skdaccess.geo.imsdnhs.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Fetches data for the Interactive Multisensor Snow and Ice Mapping System Daily Northern Hemisphere Snow and Ice Analysis.

5.46 skdaccess.geo.magnetometer Namespace Reference

Namespaces

- [data_fetcher](#)

5.47 skdaccess.geo.magnetometer.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data fetcher for USGS geomagnetic observatories.

5.48 skdaccess.geo.mahali Namespace Reference

Namespaces

- [rinex](#)
- [tec](#)
- [temperature](#)

5.49 skdaccess.geo.mahali.rinex Namespace Reference

Namespaces

- [data_fetcher](#)
- [data_wrapper](#)

5.50 skdaccess.geo.mahali.rinex.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for Mahali Data.

5.51 skdaccess.geo.mahali.rinex.data_wrapper Namespace Reference

Classes

- class [DataWrapper](#)
Data wrapper for Mahali data.

5.52 skdaccess.geo.mahali.tec Namespace Reference

Namespaces

- [data_fetcher](#)

5.53 skdaccess.geo.mahali.tec.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for Mahali Data.

5.54 skdaccess.geo.mahali.temperature Namespace Reference

Namespaces

- [data_fetcher](#)

5.55 skdaccess.geo.mahali.temperature.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for Mahali temperature data.

5.56 skdaccess.geo.modis Namespace Reference

Namespaces

- [cache](#)
- [stream](#)

5.57 skdaccess.geo.modis.cache Namespace Reference

Namespaces

- [cloud_mask](#)
- [cloud_opacity](#)
- [data_fetcher](#)
- [reflectance](#)

5.58 skdaccess.geo.modis.cache.cloud_mask Namespace Reference

Namespaces

- [data_fetcher](#)

5.59 skdaccess.geo.modis.cache.cloud_mask.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for MODIS Cloud Mask.

5.60 skdaccess.geo.modis.cache.cloud_opacity Namespace Reference

Namespaces

- [data_fetcher](#)

5.61 skdaccess.geo.modis.cache.cloud_opacity.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for MODIS Cloud Opacity.

5.62 skdaccess.geo.modis.cache.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for MODIS data.

5.63 skdaccess.geo.modis.cache.reflectance Namespace Reference

Namespaces

- [data_fetcher](#)

5.64 skdaccess.geo.modis.cache.reflectance.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

5.65 skdaccess.geo.modis.stream Namespace Reference

Namespaces

- [cloud_mask](#)
- [cloud_opacity](#)
- [data_fetcher](#)
- [reflectance](#)

5.66 skdaccess.geo.modis.stream.cloud_mask Namespace Reference

Namespaces

- [data_fetcher](#)

5.67 skdaccess.geo.modis.stream.cloud_mask.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for MODIS Cloud Mask.

5.68 skdaccess.geo.modis.stream.cloud_opacity Namespace Reference

Namespaces

- [data_fetcher](#)

5.69 skdaccess.geo.modis.stream.cloud_opacity.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for MODIS Cloud Opacity.

5.70 skdaccess.geo.modis.stream.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for MODIS data.

5.71 skdaccess.geo.modis.stream.reflectance Namespace Reference

Namespaces

- [data_fetcher](#)

5.72 skdaccess.geo.modis.stream.reflectance.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

5.73 skdaccess.geo.ngl_gps Namespace Reference

Namespaces

- [data_fetcher](#)

5.74 skdaccess.geo.ngl_gps.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data fetcher for GPS data from Nevada Geodetic Laboratory.

5.75 skdaccess.geo.pbo Namespace Reference

Namespaces

- [data_fetcher](#)

5.76 skdaccess.geo.pbo.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data fetcher for PBO GPS data.

5.77 skdaccess.geo.sentinel_1 Namespace Reference

Namespaces

- [cache](#)

5.78 skdaccess.geo.sentinel_1.cache Namespace Reference

Namespaces

- [data_fetcher](#)

5.79 skdaccess.geo.sentinel_1.cache.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
[DataFetcher](#) for retrieving Sentinel SLC data.

5.80 skdaccess.geo.srtm Namespace Reference

Namespaces

- [cache](#)

5.81 skdaccess.geo.srtm.cache Namespace Reference

Namespaces

- [data_fetcher](#)

5.82 skdaccess.geo.srtm.cache.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
[DataFetcher](#) for retrieving data from the Shuttle Radar Topography Mission.

5.83 skdaccess.geo.uavsar Namespace Reference

Namespaces

- [cache](#)

5.84 skdaccess.geo.uavsar.cache Namespace Reference

Namespaces

- [data_fetcher](#)

5.85 skdaccess.geo.uavsar.cache.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for UAVSAR data.

5.86 skdaccess.geo.wyoming_sounding Namespace Reference

Namespaces

- [cache](#)
- [stream](#)

5.87 skdaccess.geo.wyoming_sounding.cache Namespace Reference

Namespaces

- [data_fetcher](#)

5.88 skdaccess.geo.wyoming_sounding.cache.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
[DataFetcher](#) for retrieving Wyoming Sounding data.

5.89 skdaccess.geo.wyoming_sounding.stream Namespace Reference

Namespaces

- [data_fetcher](#)

5.90 skdaccess.geo.wyoming_sounding.stream.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
[DataFetcher](#) for retrieving Wyoming Sounding data.

5.91 skdaccess.planetary Namespace Reference

Namespaces

- [ode](#)

5.92 skdaccess.planetary.ode Namespace Reference

Namespaces

- [cache](#)

5.93 skdaccess.planetary.ode.cache Namespace Reference

Namespaces

- [data_fetcher](#)

5.94 skdaccess.planetary.ode.cache.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher from the Orbital Data Explorer (ODE)

5.95 skdaccess.solar Namespace Reference

Namespaces

- [sdo](#)

5.96 skdaccess.solar.sdo Namespace Reference

Namespaces

- [data_fetcher](#)

5.97 skdaccess.solar.sdo.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)
Data Fetcher for the Solar Dynamics Observatory.

5.98 skdaccess.utilities Namespace Reference

Namespaces

- [file_browser](#)
- [file_util](#)
- [grace_util](#)
- [gw_util](#)
- [image_util](#)
- [kepler_util](#)
- [mahali_util](#)
- [modis_util](#)
- [ode_util](#)
- [pbo_util](#)
- [sentinel_1_util](#)
- [sounding_util](#)
- [srtm_util](#)
- [support](#)
- [tess_utils](#)
- [uavsar_util](#)

5.99 skdaccess.utilities.file_browser Namespace Reference

Classes

- class [FileBrowser](#)

5.100 skdaccess.utilities.file_util Namespace Reference

Functions

- def [openPandasHDFStoreLocking](#) (filename, mode)
Open a pandas HDF store that may be locked:

5.100.1 Function Documentation

5.100.1.1 openPandasHDFStoreLocking()

```
def skdaccess.utilities.file_util.openPandasHDFStoreLocking (
    filename,
    mode )
```

Open a pandas HDF store that may be locked:

Parameters

<i>filename</i>	Name of file
<i>mode</i>	Mode (Such as read only, see Panda's documentation for flags)

Returns

Panda HDF store

5.101 skdaccess.utilities.grace_util Namespace Reference

Functions

- def [averageDates](#) (dates, round_nearest_day=False)
Compute the average of a pandas series of timestamps.
- def [dateMismatch](#) (dates, days=10)

Check if dates are not within a certain number of days of each other.

- def `computeEWD` (grace_data, scale_factor, round_nearest_day=False)

Compute scale corrected equivalent water depth.

- def `readTellusData` (filename, lat_lon_list, lat_name, lon_name, data_name, data_label=None, time_name=None, lat_bounds_name=None, lon_bounds_name=None, uncertainty_name=None, lat_bounds=None, lon_bounds=None)

This function reads in netcdf data provided by GRACE Tellus.

- def `getStartEndDate` (in_data)

5.101.1 Function Documentation

5.101.1.1 `averageDates()`

```
def skdaccess.utilities.grace_util.averageDates (
    dates,
    round_nearest_day = False )
```

Compute the average of a pandas series of timestamps.

Parameters

<i>dates</i>	Pandas series of pandas datetime objects
<i>round_nearest_day</i>	Round to the nearest day

Returns

Average of dates

5.101.1.2 `computeEWD()`

```
def skdaccess.utilities.grace_util.computeEWD (
    grace_data,
    scale_factor,
    round_nearest_day = False )
```

Compute scale corrected equivalent water depth.

Equivalent water depth by averaging results from GFZ, CSR, and JPL, and then applying the scale factor

Parameters

<i>grace_data</i>	Data frame containing grace data
<i>scale_factor</i>	Scale factor to apply
<i>round_nearest_day</i>	Round dates to nearest day

Returns

Equivalent water depth determined by applying the scale factor to the average GFZ, JPL and CSR.

5.101.1.3 dateMismatch()

```
def skdaccess.utilities.grace_util.dateMismatch (
    dates,
    days = 10 )
```

Check if dates are not within a certain number of days of each other.

Parameters

<i>dates</i>	Iterable container of pandas timestamps
<i>days</i>	Number of days

Returns

true if they are not with 10 days, false otherwise

5.101.1.4 getStartDate()

```
def skdaccess.utilities.grace_util.getStartDate (
    in_data )
```

5.101.1.5 readTellusData()

```
def skdaccess.utilities.grace_util.readTellusData (
    filename,
    lat_lon_list,
    lat_name,
```

```

lon_name,
data_name,
data_label = None,
time_name = None,
lat_bounds_name = None,
lon_bounds_name = None,
uncertainty_name = None,
lat_bounds = None,
lon_bounds = None )

```

This function reads in netcdf data provided by GRACE Tellus.

Parameters

<i>filename</i>	Name of file to read in
<i>lat_lon_list</i>	List of latitude, longitude tuples that are to be read
<i>data_label</i>	Label for data
<i>lat_name</i>	Name of latitude data
<i>lon_name</i>	Name of longitude data
<i>data_name</i>	Name of data product
<i>time_name</i>	Name of time data
<i>lat_bounds_name</i>	Name of latitude boundaries
<i>lon_bounds_name</i>	Name of longitude boundaries
<i>uncertainty_name</i>	Name of uncertainty in data set
<i>lat_bounds</i>	Latitude bounds
<i>lon_bounds</i>	Longitude bounds

Returns

dictionary containing data and dictionary containing latitude and longitude

5.102 skdaccess.utilities.gw_util Namespace Reference

Functions

- def [combine_water_heights](#) (in_data)
Combine median and average water heights.

5.102.1 Function Documentation

5.102.1.1 combine_water_heights()

```
def skdaccess.utilities.gw_util.combine_water_heights (  
    in_data )
```

Combine median and average water heights.

Create a column of water heights in input data frame using Median Water Depth by default, but fills in missing data using average values

Parameters

<code>in_data</code>	Input water heights data
----------------------	--------------------------

5.103 skdaccess.utilities.image_util Namespace Reference

Classes

- class [AffineGlobalCoords](#)
Convert between projected and pixel coordinates using an affine transformation.
- class [LinearGeolocation](#)
This class provides functions to convert between pixel and geodetic coordinates.
- class [SplineLatLon](#)
Holds a 2d spline for interpolating lat/lon grid.

Functions

- def [SplineGeolocation](#) (object)
This class holds splines to convert between 2d cartesian and geodetic coordinates.
- def [getExtentsFromCentersPlateCarree](#) (westmost_pixel_lon, eastmost_pixel_lon, southmost_pixel_lat, northmost_pixel_lat, lon_grid_spacing, lat_grid_spacing)
- def [convertBinCentersToEdges](#) (bin_centers, dtype=None)
Calculate edges of a set of bins from their centers.
- def [getGeoTransform](#) (extents, x_size, y_size, y_flipped=True)
Get 6 geotransform coefficients from the extents of an image and its shape.

Variables

- [x_offset](#)
- [y_offset](#)
- [lat_spline](#)
- [lon_spline](#)
- [x_spline](#)
- [y_spline](#)

5.103.1 Function Documentation

5.103.1.1 convertBinCentersToEdges()

```
def skdaccess.utilities.image_util.convertBinCentersToEdges (
    bin_centers,
    dtype = None )
```

Calculate edges of a set of bins from their centers.

Parameters

<i>bin_centers</i>	Array of bin centers
<i>dtype</i>	Data type of array used to store bin edges

Returns

bin_edges

5.103.1.2 getExtentsFromCentersPlateCarree()

```
def skdaccess.utilities.image_util.getExtentsFromCentersPlateCarree (
    westmost_pixel_lon,
    eastmost_pixel_lon,
    southmost_pixel_lat,
    northmost_pixel_lat,
    lon_grid_spacing,
    lat_grid_spacing )
```

5.103.1.3 getGeoTransform()

```
def skdaccess.utilities.image_util.getGeoTransform (
    extents,
    x_size,
    y_size,
    y_flipped = True )
```

Get 6 geotransform coefficients from the extents of an image and its shape.

Assumes origin is in the upper left and the x pixel coordinate does not depend on y projected coordinate, and the y pixel coordinate doesn't depend on the x projected coordinate

Parameters

<i>extents</i>	Image extents (x_min, x_max, y_min, y_max)
<i>x_size</i>	Number of x pixels
<i>y_size</i>	Number of y pixels
<i>y_flipped</i>	The y pixel coordinates are flipped relative to the projected coordinates

Returns

list containing the 6 affine transformation coordinates

5.103.1.4 SplineGeolocation()

```
def skdaccess.utilities.image_util.SplineGeolocation (
    object )
```

This class holds splines to convert between 2d cartesian and geodetic coordinates.

5.103.2 Variable Documentation

5.103.2.1 lat_spline

```
skdaccess.utilities.image_util.lat_spline
```

5.103.2.2 lon_spline

```
skdaccess.utilities.image_util.lon_spline
```

5.103.2.3 x_offset

```
skdaccess.utilities.image_util.x_offset
```

5.103.2.4 x_spline

```
skdaccess.utilities.image_util.x_spline
```

5.103.2.5 y_offset

```
skdaccess.utilities.image_util.y_offset
```

5.103.2.6 y_spline

```
skdaccess.utilities.image_util.y_spline
```

5.104 skdaccess.utilities.kepler_util Namespace Reference

Functions

- def [normalize](#) (in_data, column='PDCSAP_FLUX', group_column='QUARTER')
This function normalizes PDCSAP_FLUX data by quarter by dividing the flux of each quarter by the median of that respective quarter.

5.104.1 Function Documentation

5.104.1.1 normalize()

```
def skdaccess.utilities.kepler_util.normalize (
    in_data,
    column = 'PDCSAP_FLUX',
    group_column = 'QUARTER' )
```

This function normalizes PDCSAP_FLUX data by quarter by dividing the flux of each quarter by the median of that respective quarter.

Parameters

<i>in_data</i>	Pandas Data Frame to be normalized
<i>column</i>	Name of column to be normalized
<i>group_column</i>	Name of column used to group data

5.105 skdaccess.utilities.mahali_util Namespace Reference

Functions

- def [convert_date](#) (in_date)
Converts input string to pandas date time, ignores other types of objects.
- def [parselonoFile](#) (in_file, compression='infer')

5.105.1 Function Documentation

5.105.1.1 `convert_date()`

```
def skdaccess.utilities.mahali_util.convert_date (
    in_date )
```

Converts input string to pandas date time, ignores other types of objects.

Parameters

<code>in_date</code>	Input date
----------------------	------------

return pandas data time object

5.105.1.2 `parseIonoFile()`

```
def skdaccess.utilities.mahali_util.parseIonoFile (
    in_file,
    compression = 'infer' )
```

5.106 `skdaccess.utilities.modis_util` Namespace Reference

Classes

- class [LatLon](#)
Calculates Lat/Lon position from y,x pixel coordinate.

Functions

- def [getImageType](#) (in_data)
Determine what type of modis data is being processed.
- def [calibrateModis](#) (data, metadata)
This function calibrates input modis data.
- def [rescale](#) (in_array, max_val=0.9, min_val=-0.01)
This function rescales an image to fall between 0 and 1.
- def [checkBit](#) (data, bit)
Get the bit value from a bit flag.
- def [createGrid](#) (data, y_start, y_end, x_start, x_end, y_grid, x_grid, dtype, grid_fill=np.nan)
Subsets image data into a smaller image.

- def [getFileIDs](#) (modis_identifier, start_date, end_date, lat, lon, daynightboth)
Retrieve file IDs for images matching search parameters.
- def [getFileURLs](#) (file_ids)
Retrieve the ftp location for a list of file IDs.
- def [getModisData](#) (dataset, variable_name)
Loads modis data.
- def [readMODISData](#) (modis_list, variables, grid, grid_fill, use_long_name, platform, product_id)
Retrieve a list of modis data.

5.106.1 Function Documentation

5.106.1.1 [calibrateModis\(\)](#)

```
def skdaccess.utilities.modis_util.calibrateModis (  
    data,  
    metadata )
```

This function calibrates input modis data.

Parameters

<i>data</i>	Input modis data
<i>metadata</i>	Metadata associated with modis input data

Returns

calibrated modis data

5.106.1.2 [checkBit\(\)](#)

```
def skdaccess.utilities.modis_util.checkBit (  
    data,  
    bit )
```

Get the bit value from a bit flag.

Parameters

<i>data</i>	Integer bit flag
<i>bit</i>	Which bit to select (start indexing at 0)

Returns

value of chosen bit in bit flag

5.106.1.3 createGrid()

```
def skdaccess.utilities.modis_util.createGrid (
    data,
    y_start,
    y_end,
    x_start,
    x_end,
    y_grid,
    x_grid,
    dtype,
    grid_fill = np.nan )
```

Subsets image data into a smaller image.

Takes care to make sure the resulting subsection has the expected size by filling in missing data

Parameters

<i>data</i>	Input data
<i>y_start</i>	Starting pixel for y
<i>y_end</i>	Ending pixel for y
<i>x_start</i>	Starting pixel x
<i>x_end</i>	Ending pixel for x
<i>y_grid</i>	Grid size for y
<i>x_grid</i>	Grid size for x
<i>dtype</i>	The dtype of the new grid data
<i>grid_fill</i>	Fill value to use when there is no data

Returns

image subsection, fraction of valid data

5.106.1.4 getFileIDs()

```
def skdaccess.utilities.modis_util.getFileIDs (
    modis_identifier,
```

```
start_date,  
end_date,  
lat,  
lon,  
daynightboth )
```

Retrieve file IDs for images matching search parameters.

Parameters

<i>modis_identifier</i>	Product identifier (e.g. MOD09)
<i>start_date</i>	Starting date
<i>end_date</i>	Ending date
<i>lat</i>	Latitude
<i>lon</i>	Longitude
<i>daynightboth</i>	Get daytime images ('D'), nighttime images ('N') or both ('B')

Returns

list of file IDs

5.106.1.5 getFileURLs()

```
def skdaccess.utilities.modis_util.getFileURLs (  
    file_ids )
```

Retrieve the ftp location for a list of file IDs.

Parameters

<i>file_ids</i>	List of file IDs
-----------------	------------------

Returns

List of ftp locations

5.106.1.6 getImageType()

```
def skdaccess.utilities.modis_util.getImageType (  
    in_data )
```

Determine what type of modis data is being processed.

There are 3 array shapes we deal with:

```

mode 1 -> (y, x, z)
mode 2 -> (y, x)
mode 3 -> (z, y, x)

```

where z axis represents different data products and y and x correspond to the y and x image coordinates from the modis instrument

Parameters

<i>in_data</i>	Input modis data
----------------	------------------

Returns

type of modis data

5.106.1.7 getModisData()

```

def skdaccess.utilities.modis_util.getModisData (
    dataset,
    variable_name )

```

Loads modis data.

Parameters

<i>dataset</i>	netCDF4 dataset
<i>variable_name</i>	Name of variable to extract from dataset

Returns

(modis_data, metadata)

5.106.1.8 readMODISData()

```

def skdaccess.utilities.modis_util.readMODISData (
    modis_list,
    variables,
    grid,
    grid_fill,
    use_long_name,
    platform,
    product_id )

```

Retrieve a list of modis data.

Parameters

<i>modis_list</i>	List of MODIS data to load
<i>variables</i>	List of variables in the MODIS data to load
<i>grid</i>	Further divide each image into a multiple grids of size (y,x)
<i>grid_fill</i>	Fill value to use when creating gridded data
<i>use_long_name</i>	Use long names for metadata instead of variable name
<i>platform</i>	Which satellite to use, either MOD or MYD.
<i>product_id</i>	Product string (e.g. '06_L2')

5.106.1.9 rescale()

```
def skdaccess.utilities.modis_util.rescale (
    in_array,
    max_val = 0.9,
    min_val = -0.01 )
```

This function rescales an image to fall between 0 and 1.

Parameters

<i>in_array</i>	Data to be rescaled
<i>max_val</i>	Values greater than or equal to max_val will become 1
<i>min_val</i>	Values less than or equal to min_val will become 0

Returns

scaled data

5.107 skdaccess.utilities.ode_util Namespace Reference

Functions

- def [query_yes_no](#) (question, default="yes")
 - def [get_query_url](#) (target, mission, instrument, product_type, western_lon, eastern_lon, min_lat, max_lat, min_ob_time, max_ob_time, product_id, query_type, output, results, number_product_limit, result_offset_number)
 - def [get_files_urls](#) (query_url, file_name='*', print_info=False)
 - def [query_files_urls](#) (target, mission, instrument, product_type, western_lon, eastern_lon, min_lat, max_lat, min_ob_time, max_ob_time, product_id, file_name, number_product_limit, result_offset_number)
- Retrieve the URL locations based on a query using ODE REST interface.*
- def [correct_CRISM_label](#) (label_file_location)
 - def [correct_file_name_case_in_label](#) (label_file_location, other_file_locations)

- def [correct_label_file](#) (label_file_location, other_file_locations=[])
Correct a label file if GDAL cannot open the corresponding data file.
- def [get_raster_array](#) (gdal_raster, remove_ndv=True)
Get a NumPy array from a raster opened with GDAL.
- def [get_raster_extent](#) (gdal_raster)
Get the extent of a raster opened with GDAL.

5.107.1 Function Documentation

5.107.1.1 [correct_CRISM_label\(\)](#)

```
def skdaccess.utilities.ode_util.correct_CRISM_label (
    label_file_location )
```

5.107.1.2 [correct_file_name_case_in_label\(\)](#)

```
def skdaccess.utilities.ode_util.correct_file_name_case_in_label (
    label_file_location,
    other_file_locations )
```

5.107.1.3 [correct_label_file\(\)](#)

```
def skdaccess.utilities.ode_util.correct_label_file (
    label_file_location,
    other_file_locations = [] )
```

Correct a label file if GDAL cannot open the corresponding data file.

Parameters

<i>label_file_location</i>	Local address of the current label
<i>other_file_locations</i>	Other files that were downloaded with the label file

Returns

Local address of the new label

5.107.1.4 `get_files_urls()`

```
def skdaccess.utilities.ode_util.get_files_urls (
    query_url,
    file_name = '*',
    print_info = False )
```

5.107.1.5 `get_query_url()`

```
def skdaccess.utilities.ode_util.get_query_url (
    target,
    mission,
    instrument,
    product_type,
    western_lon,
    eastern_lon,
    min_lat,
    max_lat,
    min_ob_time,
    max_ob_time,
    product_id,
    query_type,
    output,
    results,
    number_product_limit,
    result_offset_number )
```

5.107.1.6 `get_raster_array()`

```
def skdaccess.utilities.ode_util.get_raster_array (
    gdal_raster,
    remove_ndv = True )
```

Get a NumPy array from a raster opened with GDAL.

Parameters

<i>gdal_raster</i>	A raster opened with GDAL
<i>remove_ndv</i>	Replace the no-data value as mentionned in the label by np.nan

Returns

The array

5.107.1.7 `get_raster_extent()`

```
def skdaccess.utilities.ode_util.get_raster_extent (
    gdal_raster )
```

Get the extent of a raster opened with GDAL.

Parameters

<i>gdal_raster</i>	A raster opened with GDAL
--------------------	---------------------------

Returns

The raster extent

5.107.1.8 `query_files_urls()`

```
def skdaccess.utilities.ode_util.query_files_urls (
    target,
    mission,
    instrument,
    product_type,
    western_lon,
    eastern_lon,
    min_lat,
    max_lat,
    min_ob_time,
    max_ob_time,
    product_id,
    file_name,
    number_product_limit,
    result_offset_number )
```

Retrieve the URL locations based on a query using ODE REST interface.

Parameters

<i>target</i>	Aimed planetary body, i.e., Mars, Mercury, Moon, Phobos, or Venus
<i>mission</i>	Aimed mission, e.g., MGS or MRO
<i>instrument</i>	Aimed instrument from the mission, e.g., HIRISE or CRISM
<i>product_type</i>	Type of product to look for, e.g., DTM or RDRV11
<i>western_lon</i>	Western longitude to look for the data, from 0 to 360
<i>eastern_lon</i>	Eastern longitude to look for the data, from 0 to 360
<i>min_lat</i>	Minimal latitude to look for the data, from -90 to 90
<i>max_lat</i>	Maximal latitude to look for the data, from -90 to 90
<i>min_ob_time</i>	Minimal observation time in (even partial) UTC format, e.g., '2017-03-01'

Parameters

<i>max_ob_time</i>	Maximal observation time in (even partial) UTC format, e.g., '2017-03-01'
<i>product_id</i>	PDS Product Id to look for, with wildcards (*) allowed
<i>file_name</i>	File name to look for, with wildcards (*) allowed
<i>number_product_limit</i>	Maximal number of products to return (100 at most)
<i>result_offset_number</i>	Offset the return products, to go beyond the limit of 100 returned products

Returns

List of URL locations

5.107.1.9 query_yes_no()

```
def skdaccess.utilities.ode_util.query_yes_no (
    question,
    default = "yes" )
```

5.108 skdaccess.utilities.pbo_util Namespace Reference

Functions

- def [getStationCoords](#) (pbo_info, station_list)
Get the station coordinates for a list of stations.
- def [getLatLonRange](#) (pbo_info, station_list)
Retrive the range of latitude and longitude occupied by a set of stations.
- def [getROIstations](#) (geo_point, radiusParam, data, header)
This function returns the 4ID station codes for the stations in a region.
- def [stab_sys](#) (data_iterator, metadata, stab_min_NE=.0005, stab_min_U=.005, sigsc=2, errProp=1)
Stabilize GPS data to a region.
- def [propagateErrors](#) (R, sc, stationCovs)
Propagate GPS errors.
- def [nostab_sys](#) (allH, allD, timerng, indx=1, mdyratio=.7, use_progress_bar=True, index_date_only=False)
Do not apply stabilization and simply returns stations after checking for sufficient amount of data.
- def [removeAntennaOffset](#) (antenna_offsets, data, window_start=pd.to_timedelta('4D'), window_end=pd.to_←timedelta('4D'), min_diff=0.005, debug=False)
Remove offsets caused by changes in antennas.

5.108.1 Function Documentation

5.108.1.1 getLatLonRange()

```
def skdaccess.utilities.pbo_util.getLatLonRange (
    pbo_info,
    station_list )
```

Retrive the range of latitude and longitude occupied by a set of stations.

Parameters

<i>pbo_info</i>	PBO Metadata
<i>station_list</i>	List of stations

Returns

list containg two tuples, lat_range and lon_range

5.108.1.2 getROIstations()

```
def skdaccess.utilities.pbo_util.getROIstations (
    geo_point,
    radiusParam,
    data,
    header )
```

This function returns the 4ID station codes for the stations in a region.

The region of interest is defined by the geographic coordinate and a window size

Parameters

<i>geo_point</i>	The geographic (lat,lon) coordinate of interest
<i>radiusParam</i>	An overloaded radius of interest [km] or latitude and longitude window [deg] around the geo_point
<i>data</i>	Stabilized (or unstabilized) data generated from the data fetcher or out of stab_sys
<i>header</i>	Header dictionary with stations metadata keyed by their 4ID code. This is output with the data.

Returns

station_list, list of site 4ID codes in the specified geographic region

5.108.1.3 getStationCoords()

```
def skdaccess.utilities.pbo_util.getStationCoords (
    pbo_info,
    station_list )
```

Get the station coordinates for a list of stations.

Parameters

<i>pbo_info</i>	PBO Metadata
<i>station_list</i>	List of stations

Returns

list of tuples containing lat, lon coordinates of stations

5.108.1.4 nostab_sys()

```
def skdaccess.utilities.pbo_util.nostab_sys (
    allH,
    allD,
    timerng,
    indx = 1,
    mdyratio = .7,
    use_progress_bar = True,
    index_date_only = False )
```

Do not apply stabilization and simply returns stations after checking for sufficient amount of data.

Parameters

<i>allH</i>	a dictionary of all of the headers of all sites loaded from the data directory
<i>allD</i>	a dictionary of all of the panda format data of all of the corresponding sites
<i>timerng</i>	an array with two string elements, describing the starting and ending dates
<i>indx</i>	a list of site 4ID's indicating stations in the relevant geographic location, or 1 for all sites
<i>mdyratio</i>	optional parameter for the minimum required ratio of data to determine if a site is kept for further analysis
<i>use_progress_bar</i>	Display a progress bar
<i>index_date_only</i>	When creating an index for the data, use date (not the time) only

Returns

smSet, a reduced size dictionary of the data (in meters) for the sites in the specified geographic region and smHdr, a reduced size dictionary of the headers for the sites in the region

5.108.1.5 propagateErrors()

```
def skdaccess.utilities.pbo_util.propagateErrors (
    R,
    sc,
    stationCovs )
```

Propagate GPS errors.

By writing out the $R \cdot E \cdot R.T$ equations... to calculate the new covariance matrix without needing to form the matrix first as an intermediate step. Modifies covariance matrix in place

Parameters

<i>R</i>	Rotation matrix
<i>sc</i>	Scaling value
<i>stationCovs</i>	Station Covariances

5.108.1.6 removeAntennaOffset()

```
def skdaccess.utilities.pbo_util.removeAntennaOffset (
    antenna_offsets,
    data,
    window_start = pd.to_timedelta('4D'),
    window_end = pd.to_timedelta('4D'),
    min_diff = 0.005,
    debug = False )
```

Remove offsets caused by changes in antennas.

Parameters

<i>antenna_offsets</i>	Pandas series of dates describing when the antenna changes were made
<i>data</i>	Input GPS data
<i>window_start</i>	Starting time before and after event to use for calculating offset
<i>window_end</i>	Ending time before and after event to use before calculating offset
<i>min_diff</i>	Minimum difference before and after offset to for applying correction
<i>debug</i>	Enable debug output

Returns

GPS data with the offsets removed

5.108.1.7 stab_sys()

```
def skdaccess.utilities.pbo_util.stab_sys (
    data_iterator,
    metadata,
    stab_min_NE = .0005,
    stab_min_U = .005,
    sigsc = 2,
    errProp = 1 )
```

Stabilize GPS data to a region.

The stab_sys function is a Python implementation of the Helmert 7-parameter transformation, used to correct for common mode error. This builds on Prof Herring's stab_sys function in his tscon Fortran code. It uses a SVD approach to estimating the rotation matrix gathered from 'Computing Helmert Transformations' by G.A. Watson as well as its references. Note that units should be in meters, that is in the format from the level 2 processed UNAVCO pos files

Parameters

<i>data_iterator</i>	Expects an iterator that returns label, pandas dataframe
<i>metadata</i>	Metadata that contains 'refXYZ' and 'refNEU'
<i>stab_min_NE</i>	Optional minimum horizontal covariance parameter
<i>stab_min_U</i>	Optional minimum vertical covariance parameter
<i>sigsc</i>	Optional scaling factor for determining cutoff bounds for non stable sites
<i>errProp</i>	Propagate errors through the transformation

Returns

smSet, a reduced size dictionary of the data (in mm) for the sites in the specified geographic region, smHdr, a reduced size dictionary of the headers for the sites in the region

5.109 skdaccess.utilities.sentinel_1_util Namespace Reference

Functions

- def [parseSatelliteData](#) (in_satellite_file)
Parse Sentinel satellite data.

5.109.1 Function Documentation

5.109.1.1 parseSatelliteData()

```
def skdaccess.utilities.sentinel_1_util.parseSatelliteData (
    in_satellite_file )
```

Parse Sentinel satellite data.

Parameters

<code>in_satellite_file</code>	Satellite orbit filename
--------------------------------	--------------------------

Returns

DataFrame of orbit information

5.110 skdaccess.utilities.sounding_util Namespace Reference

Classes

- class [SoundingParser](#)
This class parses Wyoming Sounding data.

Functions

- def [generateQueries](#) (station_number, year_list, month_list, day_start, day_end, start_hour, end_hour)
Generate url queries for sounding data.

5.110.1 Function Documentation

5.110.1.1 generateQueries()

```
def skdaccess.utilities.sounding_util.generateQueries (
    station_number,
    year_list,
    month_list,
    day_start,
    day_end,
    start_hour,
    end_hour )
```

Generate url queries for sounding data.

Parameters

<i>station_number</i>	Input station number
<i>year_list</i>	Input years as a list
<i>month_list</i>	Input month as a list
<i>day_start</i>	Starting day
<i>day_end</i>	Ending day
<i>start_hour</i>	Starting hour
<i>end_hour</i>	Ending hour

Returns

list of urls containing requested data

5.111 skdaccess.utilities.srtm_util Namespace Reference

Functions

- def [merge_srtm_tiles](#) (srtm_tiles, lon_min, lon_max, lat_min, lat_max)
- def [getSRTMLatLon](#) (lat_min, lat_max, lon_min, lon_max)
Retrieve parameters that encompass area when creating SRTM data fetcher.
- def [getSRTMData](#) (srtmdw, lat_start, lat_end, lon_start, lon_end)
Select SRTM data in a latitude/longitude box.

5.111.1 Function Documentation

5.111.1.1 [getSRTMData\(\)](#)

```
def skdaccess.utilities.srtm_util.getSRTMData (
    srtmdw,
    lat_start,
    lat_end,
    lon_start,
    lon_end )
```

Select SRTM data in a latitude/longitude box.

Parameters

<i>srtmdw</i>	SRTM data wrapper
<i>lat_start</i>	Starting latiude
<i>lat_end</i>	Ending latiude
<i>lon_start</i>	Starting longitude
<i>lon_end</i>	Ending longitude
<i>flip_y</i>	Flip the y axis so that increasing y pixels are increasing in latitude

Returns

Tuple containing the cut data, new extents, and a affine geotransform coefficients

5.111.1.2 getSRTMLatLon()

```
def skdaccess.utilities.srtm_util.getSRTMLatLon (
    lat_min,
    lat_max,
    lon_min,
    lon_max )
```

Retrieve parameters that encompass area when creating SRTM data fetcher.

Parameters

<i>lat_min</i>	Minimum latitude
<i>lat_max</i>	Maximum latitude
<i>lon_min</i>	Minimum longitude
<i>lon_max</i>	Maximum longitude

Returns

(starting_latitude, ending_latitude, starting_longitude, ending_longitude)

5.111.1.3 merge_srtm_tiles()

```
def skdaccess.utilities.srtm_util.merge_srtm_tiles (
    srtm_tiles,
    lon_min,
    lon_max,
    lat_min,
    lat_max )
```

5.112 skdaccess.utilities.support Namespace Reference**Functions**

- def [retrieveCommonDatesHDF](#) (support_data_filename, key_list, in_date_list)
Get a list of all dates that have data available.
- def [progress_bar](#) (in_iterable, total=None, enabled=True)
Progress bar using tqdm.
- def [convertToStr](#) (in_value, zfill=0)
- def [join_string](#) (part1, part2, concatenation_string='AND', seperator=' ')
Join two strings together using a concatenation string.

5.112.1 Function Documentation

5.112.1.1 convertToStr()

```
def skdaccess.utilities.support.convertToStr (
    in_value,
    zfill = 0 )
```

5.112.1.2 join_string()

```
def skdaccess.utilities.support.join_string (
    part1,
    part2,
    concatenation_string = 'AND',
    seperator = ' ' )
```

Join two strings together using a concatenation string.

Handles the case where either part1 or part2 are an empty string

Parameters

<i>part1</i>	First string
<i>part2</i>	Second string
<i>concatenation_string</i>	String used to join part1 and part2
<i>seperator</i>	Seperator used to between each part and the concatenation string

Returns

A single string that consists of the part1 and part2 joined together using a concatenation string

5.112.1.3 progress_bar()

```
def skdaccess.utilities.support.progress_bar (
    in_iterable,
    total = None,
    enabled = True )
```

Progress bar using tqdm.

Parameters

<i>in_iterable</i>	Input iterable
<i>total</i>	Total number of elements
<i>enabled</i>	Enable progress bar

5.112.1.4 retrieveCommonDatesHDF()

```
def skdaccess.utilities.support.retrieveCommonDatesHDF (
    support_data_filename,
    key_list,
    in_date_list )
```

Get a list of all dates that have data available.

Parameters

<i>support_data_filename</i>	Filename of support data
<i>key_list</i>	List of keys in HDF file
<i>in_date_list</i>	Input date list to check

Returns

dictionary of dates with data

5.113 skdaccess.utilities.tess_utils Namespace Reference

Functions

- def [parseTessData](#) (fits_data)
Retrieve Tess lightcurve data from astropy.io.fits.HDUList object.

5.113.1 Function Documentation

5.113.1.1 parseTessData()

```
def skdaccess.utilities.tess_utils.parseTessData (
    fits_data )
```

Retrieve Tess lightcurve data from astropy.io.fits.HDUList object.

Parameters

<i>fits_data</i>	astropy.io.fits.HDUList object that corresponding to a Tess lightcurve fits file
------------------	--

Returns

Pandas data frame of light curve, ordered dictionary of metadata

5.114 skdaccess.utilities.uavsar_util Namespace Reference

Functions

- def [readUAVSARMetadata](#) (in_file)
Parse UAVSAR metadata.

5.114.1 Function Documentation

5.114.1.1 readUAVSARMetadata()

```
def skdaccess.utilities.uavsar_util.readUAVSARMetadata (
    in_file )
```

Parse UAVSAR metadata.

Parameters

<i>in_file</i>	String of Metadata filename or file object (file should end in .ann)
----------------	--

Returns

OrderedDict of metadata

5.115 terminal_groundwater_example Namespace Reference

Variables

- [fullIDF](#)
- [fullIDW](#) = fullIDF.output()
- [meta_data](#) = WDF.getStationMetadata()

- [dataIt = fullDW.getIterator\(\)](#)
- [label_1](#)
- [data_1](#)
- [label_2](#)
- [data_2](#)
- [color](#)

5.115.1 Variable Documentation

5.115.1.1 color

`terminal_groundwater_example.color`

5.115.1.2 data_1

`terminal_groundwater_example.data_1`

5.115.1.3 data_2

`terminal_groundwater_example.data_2`

5.115.1.4 dataIt

`terminal_groundwater_example.dataIt = fullDW.getIterator()`

5.115.1.5 fullDF

`terminal_groundwater_example.fullDF`

Initial value:

```
1 = WDF([AutoParam(35), AutoParam(38), AutoParam(-119), AutoParam(-118)],
2      '2007-01-01', '2016-12-31', cutoff=0.0)
```


5.115.1.6 fullDW

```
terminal_groundwater_example.fullDW = fullDF.output()
```

5.115.1.7 label_1

```
terminal_groundwater_example.label_1
```

5.115.1.8 label_2

```
terminal_groundwater_example.label_2
```

5.115.1.9 meta_data

```
terminal_groundwater_example.meta_data = WDF.getStationMetadata()
```

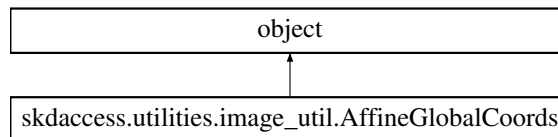

Chapter 6

Class Documentation

6.1 skdaccess.utilities.image_util.AffineGlobalCoords Class Reference

Convert between projected and pixel coordinates using an affine transformation.

Inheritance diagram for skdaccess.utilities.image_util.AffineGlobalCoords:



Public Member Functions

- `def __init__ (self, aff_coeffs, center_pixels=False)`
Initialize Global Coords Object.
- `def getProjectedYX (self, y_array, x_array)`
Convert pixel coordinates to projected coordinates.
- `def getPixelYX (self, y_proj, x_proj)`
Convert from projected coordinates to pixel coordinates.

6.1.1 Detailed Description

Convert between projected and pixel coordinates using an affine transformation.

6.1.2 Constructor & Destructor Documentation

6.1.2.1 `__init__()`

```
def skdaccess.utilities.image_util.AffineGlobalCoords.__init__ (
    self,
    aff_coeffs,
    center_pixels = False )
```

Initialize Global Coords Object.

Parameters

<i>aff_coeffs</i>	Affine coefficients
<i>center_pixels</i>	Apply offsets so that integer values refer to the center of the pixel and not the edge

6.1.3 Member Function Documentation

6.1.3.1 `getPixelYX()`

```
def skdaccess.utilities.image_util.AffineGlobalCoords.getPixelYX (
    self,
    y_proj,
    x_proj )
```

Convert from projected coordinates to pixel coordinates.

Parameters

<i>y_proj</i>	Input projected y coordinates
<i>x_proj</i>	Input projected x coordinates

Returns

y pixel coordinates, x pixel coordinates

6.1.3.2 `getProjectedYX()`

```
def skdaccess.utilities.image_util.AffineGlobalCoords.getProjectedYX (
    self,
    y_array,
    x_array )
```

Convert pixel coordinates to projected coordinates.

Parameters

<code>y_array</code>	Input y pixel coordinates
<code>x_array</code>	Input x pixel coordinates

Returns

projected y coordinates, projected x coordinates

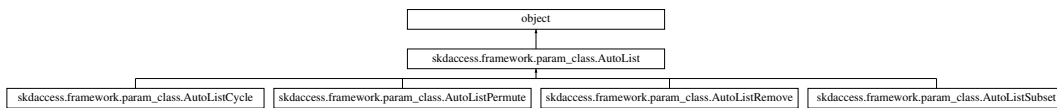
The documentation for this class was generated from the following file:

- [utilities/image_util.py](#)

6.2 skdaccess.framework.param_class.AutoList Class Reference

Specifies a list for returning selections of lists, as opposed to a single element.

Inheritance diagram for `skdaccess.framework.param_class.AutoList`:



Public Member Functions

- `def __init__(self, val_list)`
Construct a `AutoList` object.
- `def val(self)`
Retrieves current list of parameters.
- `def perturb(self)`
This class doesn't change the list when being perturbed.
- `def reset(self)`
Reset current list to initial list.
- `def getAllOptions(self)`
Get all possible options.
- `def __str__(self)`
String representation of class.
- `def __len__(self)`
Retrieves the length of parameters contained in the list.
- `def __getitem__(self, ii)`
Retrieves item from list.
- `def __setitem__(self, ii, val)`
Set a value in the list.
- `def __call__(self)`
Retrieve current list.

Public Attributes

- [val_init](#)
- [val_list](#)

6.2.1 Detailed Description

Specifies a list for returning selections of lists, as opposed to a single element.

6.2.2 Constructor & Destructor Documentation

6.2.2.1 `__init__()`

```
def skdaccess.framework.param_class.AutoList.__init__ (
    self,
    val_list )
```

Construct a [AutoList](#) object.

Parameters

<i>val_list</i>	List of parameters
-----------------	--------------------

6.2.3 Member Function Documentation

6.2.3.1 `__call__()`

```
def skdaccess.framework.param_class.AutoList.__call__ (
    self )
```

Retrieve current list.

Returns

Current list

6.2.3.2 `__getitem__()`

```
def skdaccess.framework.param_class.AutoList.__getitem__ (
    self,
    ii )
```

Retrieves item from list.

Parameters

<i>ii</i>	Index of item to be retrieved
-----------	-------------------------------

Returns

Item at index ii

6.2.3.3 `__len__()`

```
def skdaccess.framework.param_class.AutoList.__len__ (
    self )
```

Retrieves the length of parameters contained in the list.

Returns

Number of elements in the list

6.2.3.4 `__setitem__()`

```
def skdaccess.framework.param_class.AutoList.__setitem__ (
    self,
    ii,
    val )
```

Set a value in the list.

Parameters

<i>ii</i>	Index of list to be set
<i>val</i>	Input value

6.2.3.5 `__str__()`

```
def skdaccess.framework.param_class.AutoList.__str__ (
    self )
```

String representation of class.

Returns

String containing all parameters in list

6.2.3.6 `getAllOptions()`

```
def skdaccess.framework.param_class.AutoList.getAllOptions (
    self )
```

Get all possible options.

Returns

List that contains every option that could possibly be selected

6.2.3.7 `perturb()`

```
def skdaccess.framework.param_class.AutoList.perturb (
    self )
```

This class doesn't change the list when being perturbed.

6.2.3.8 `reset()`

```
def skdaccess.framework.param_class.AutoList.reset (
    self )
```

Reset current list to initial list.

6.2.3.9 val()

```
def skdaccess.framework.param_class.AutoList.val (
    self )
```

Retrieves current list of parameters.

Returns

List of current parameters

6.2.4 Member Data Documentation

6.2.4.1 val_init

```
skdaccess.framework.param_class.AutoList.val_init
```

6.2.4.2 val_list

```
skdaccess.framework.param_class.AutoList.val_list
```

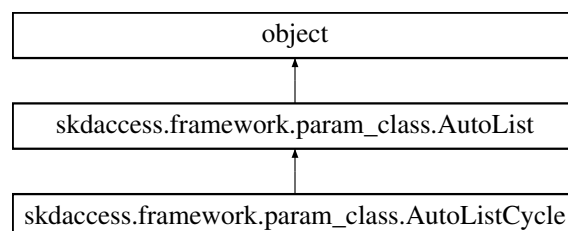
The documentation for this class was generated from the following file:

- framework/[param_class.py](#)

6.3 skdaccess.framework.param_class.AutoListCycle Class Reference

An Autolist that cycles through different lists.

Inheritance diagram for skdaccess.framework.param_class.AutoListCycle:



Public Member Functions

- def `__init__` (self, `list_val_list`)
Construct a `AutoList_Cycle` object.
- def `perturb` (self)
Select next list from list of lists.
- def `reset` (self)
Resets to the first list in the list of lists.
- def `getAllOptions` (self)
Get elements that could possibly be called.
- def `val` (self)
Retrieves current list of parameters.
- def `__str__` (self)
String representation of class.
- def `__len__` (self)
Retrieves the length of parameters contained in the list.
- def `__getitem__` (self, ii)
Retrieves item from list.
- def `__setitem__` (self, ii, `val`)
Set a value in the list.
- def `__call__` (self)
Retrieve current list.

Public Attributes

- `list_val_list`
- `val_list`
- `index`
- `val_init`

6.3.1 Detailed Description

An Autolist that cycles through different lists.

6.3.2 Constructor & Destructor Documentation

6.3.2.1 `__init__()`

```
def skdaccess.framework.param_class.AutoListCycle.__init__ (
    self,
    list_val_list )
```

Construct a `AutoList_Cycle` object.

Parameters

<i>list_val_list</i>	List of different lists to cycle through
----------------------	--

6.3.3 Member Function Documentation**6.3.3.1 __call__()**

```
def skdaccess.framework.param_class.AutoList.__call__ (
    self ) [inherited]
```

Retrieve current list.

Returns

Current list

6.3.3.2 __getitem__()

```
def skdaccess.framework.param_class.AutoList.__getitem__ (
    self,
    ii ) [inherited]
```

Retrieves item from list.

Parameters

<i>ii</i>	Index of item to be retrieved
-----------	-------------------------------

Returns

Item at index ii

6.3.3.3 __len__()

```
def skdaccess.framework.param_class.AutoList.__len__ (
    self ) [inherited]
```

Retrieves the length of parameters contained in the list.

Returns

Number of elements in the list

6.3.3.4 __setitem__()

```
def skdaccess.framework.param_class.AutoList.__setitem__ (
    self,
    ii,
    val ) [inherited]
```

Set a value in the list.

Parameters

<i>ii</i>	Index of list to be set
<i>val</i>	Input value

6.3.3.5 __str__()

```
def skdaccess.framework.param_class.AutoList.__str__ (
    self ) [inherited]
```

String representation of class.

Returns

String containing all parameters in list

6.3.3.6 getAllOptions()

```
def skdaccess.framework.param_class.AutoListCycle.getAllOptions (
    self )
```

Get elements that could possibly be called.

Returns

List of all possible elements

6.3.3.7 perturb()

```
def skdaccess.framework.param_class.AutoListCycle.perturb (  
    self )
```

Select next list from list of lists.

6.3.3.8 reset()

```
def skdaccess.framework.param_class.AutoListCycle.reset (  
    self )
```

Resets to the first list in the list of lists.

6.3.3.9 val()

```
def skdaccess.framework.param_class.AutoList.val (  
    self ) [inherited]
```

Retrieves current list of parameters.

Returns

List of current parameters

6.3.4 Member Data Documentation

6.3.4.1 index

```
skdaccess.framework.param_class.AutoListCycle.index
```

6.3.4.2 list_val_list

```
skdaccess.framework.param_class.AutoListCycle.list_val_list
```

6.3.4.3 val_init

`skdaccess.framework.param_class.AutoList.val_init` [inherited]

6.3.4.4 val_list

`skdaccess.framework.param_class.AutoListCycle.val_list`

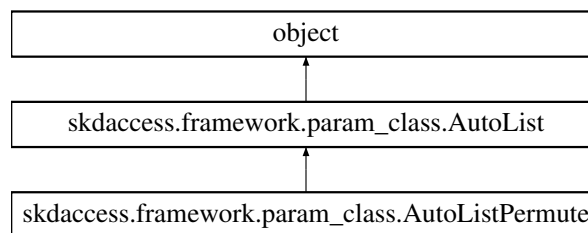
The documentation for this class was generated from the following file:

- [framework/param_class.py](#)

6.4 skdaccess.framework.param_class.AutoListPermute Class Reference

A perturber that permutes a list.

Inheritance diagram for `skdaccess.framework.param_class.AutoListPermute`:



Public Member Functions

- `def perturb (self)`
Randomly permutes the initial list.
- `def val (self)`
Retrieves current list of parameters.
- `def reset (self)`
Reset current list to initial list.
- `def getAllOptions (self)`
Get all possible options.
- `def __str__ (self)`
String representation of class.
- `def __len__ (self)`
Retrieves the length of parameters contained in the list.
- `def __getitem__ (self, ii)`
Retrieves item from list.
- `def __setitem__ (self, ii, val)`
Set a value in the list.
- `def __call__ (self)`
Retrieve current list.

Public Attributes

- [val_init](#)
- [val_list](#)

6.4.1 Detailed Description

A perturber that permutes a list.

6.4.2 Member Function Documentation

6.4.2.1 `__call__()`

```
def skdaccess.framework.param_class.AutoList.__call__ (
    self ) [inherited]
```

Retrieve current list.

Returns

Current list

6.4.2.2 `__getitem__()`

```
def skdaccess.framework.param_class.AutoList.__getitem__ (
    self,
    ii ) [inherited]
```

Retrieves item from list.

Parameters

<i>ii</i>	Index of item to be retrieved
-----------	-------------------------------

Returns

Item at index ii

6.4.2.3 `__len__()`

```
def skdaccess.framework.param_class.AutoList.__len__ (
    self ) [inherited]
```

Retrieves the length of parameters contained in the list.

Returns

Number of elements in the list

6.4.2.4 `__setitem__()`

```
def skdaccess.framework.param_class.AutoList.__setitem__ (
    self,
    ii,
    val ) [inherited]
```

Set a value in the list.

Parameters

<i>ii</i>	Index of list to be set
<i>val</i>	Input value

6.4.2.5 `__str__()`

```
def skdaccess.framework.param_class.AutoList.__str__ (
    self ) [inherited]
```

String representation of class.

Returns

String containing all parmaters in list

6.4.2.6 getAllOptions()

```
def skdaccess.framework.param_class.AutoList.getAllOptions (
    self ) [inherited]
```

Get all possible options.

Returns

List that contains every option that could possibly be selected

6.4.2.7 perturb()

```
def skdaccess.framework.param_class.AutoListPermute.perturb (
    self )
```

Randomly permutes the initial list.

6.4.2.8 reset()

```
def skdaccess.framework.param_class.AutoList.reset (
    self ) [inherited]
```

Reset current list to initial list.

6.4.2.9 val()

```
def skdaccess.framework.param_class.AutoList.val (
    self ) [inherited]
```

Retrieves current list of parameters.

Returns

List of current parameters

6.4.3 Member Data Documentation

6.4.3.1 val_init

`skdaccess.framework.param_class.AutoList.val_init` [inherited]

6.4.3.2 val_list

`skdaccess.framework.param_class.AutoList.val_list` [inherited]

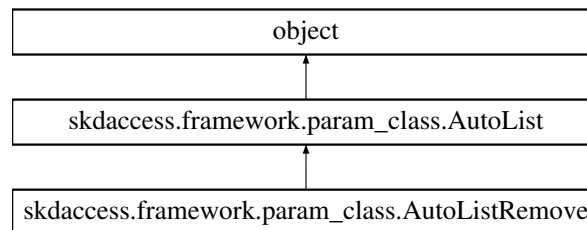
The documentation for this class was generated from the following file:

- [framework/param_class.py](#)

6.5 skdaccess.framework.param_class.AutoListRemove Class Reference

Removes a different single element from the initial list at each perturb call.

Inheritance diagram for `skdaccess.framework.param_class.AutoListRemove`:



Public Member Functions

- `def __init__ (self, val_list)`
Construct a AutoList_Cycle object.
- `def perturb (self)`
Systematically change which item is absent from the list.
- `def reset (self)`
Reset the list to its initial value.
- `def val (self)`
Retrieves current list of parameters.
- `def getAllOptions (self)`
Get all possible options.
- `def __str__ (self)`
String representation of class.
- `def __len__ (self)`
Retrieves the length of parameters contained in the list.
- `def __getitem__ (self, ii)`
Retrieves item from list.
- `def __setitem__ (self, ii, val)`
Set a value in the list.
- `def __call__ (self)`
Retrieve current list.

Public Attributes

- `n`
- `val_list`
- `val_init`

6.5.1 Detailed Description

Removes a different single element from the initial list at each perturb call.

6.5.2 Constructor & Destructor Documentation

6.5.2.1 `__init__()`

```
def skdaccess.framework.param_class.AutoListRemove.__init__ (
    self,
    val_list )
```

Construct a AutoList_Cycle object.

Parameters

<code>val_list</code>	Initial list of parameters.
-----------------------	-----------------------------

6.5.3 Member Function Documentation

6.5.3.1 `__call__()`

```
def skdaccess.framework.param_class.AutoList.__call__ (
    self ) [inherited]
```

Retrieve current list.

Returns

Current list

6.5.3.2 `__getitem__()`

```
def skdaccess.framework.param_class.AutoList.__getitem__ (
    self,
    ii ) [inherited]
```

Retrieves item from list.

Parameters

<i>ii</i>	Index of item to be retrieved
-----------	-------------------------------

Returns

Item at index *ii*

6.5.3.3 `__len__()`

```
def skdaccess.framework.param_class.AutoList.__len__ (
    self ) [inherited]
```

Retrieves the length of parameters contained in the list.

Returns

Number of elements in the list

6.5.3.4 `__setitem__()`

```
def skdaccess.framework.param_class.AutoList.__setitem__ (
    self,
    ii,
    val ) [inherited]
```

Set a value in the list.

Parameters

<i>ii</i>	Index of list to be set
<i>val</i>	Input value

6.5.3.5 `__str__()`

```
def skdaccess.framework.param_class.AutoList.__str__ (
    self ) [inherited]
```

String representation of class.

Returns

String containing all parameters in list

6.5.3.6 `getAllOptions()`

```
def skdaccess.framework.param_class.AutoList.getAllOptions (
    self ) [inherited]
```

Get all possible options.

Returns

List that contains every option that could possibly be selected

6.5.3.7 `perturb()`

```
def skdaccess.framework.param_class.AutoListRemove.perturb (
    self )
```

Systematically change which item is absent from the list.

6.5.3.8 `reset()`

```
def skdaccess.framework.param_class.AutoListRemove.reset (
    self )
```

Reset the list to its initial value.

6.5.3.9 val()

```
def skdaccess.framework.param_class.AutoList.val (
    self ) [inherited]
```

Retrieves current list of parameters.

Returns

List of current parameters

6.5.4 Member Data Documentation

6.5.4.1 n

```
skdaccess.framework.param_class.AutoListRemove.n
```

6.5.4.2 val_init

```
skdaccess.framework.param_class.AutoList.val_init [inherited]
```

6.5.4.3 val_list

```
skdaccess.framework.param_class.AutoListRemove.val_list
```

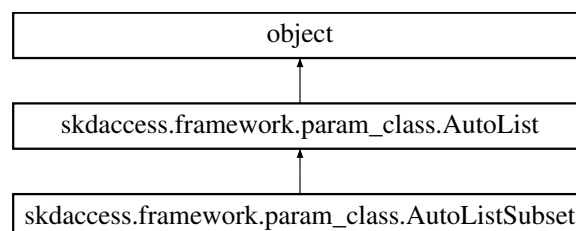
The documentation for this class was generated from the following file:

- framework/[param_class.py](#)

6.6 skdaccess.framework.param_class.AutoListSubset Class Reference

An [AutoList](#) perturber that creates random subsets of a list.

Inheritance diagram for skdaccess.framework.param_class.AutoListSubset:



Public Member Functions

- def [perturb](#) (self)
Perturb the list by selecting a random subset of the initial list.
- def [val](#) (self)
Retrieves current list of parameters.
- def [reset](#) (self)
Reset current list to initial list.
- def [getAllOptions](#) (self)
Get all possible options.
- def [__str__](#) (self)
String representation of class.
- def [__len__](#) (self)
Retrieves the length of parameters contained in the list.
- def [__getitem__](#) (self, ii)
Retrieves item from list.
- def [__setitem__](#) (self, ii, [val](#))
Set a value in the list.
- def [__call__](#) (self)
Retrieve current list.

Public Attributes

- [val_list](#)
- [val_init](#)

6.6.1 Detailed Description

An [AutoList](#) perturber that creates random subsets of a list.

List can be empty

6.6.2 Member Function Documentation

6.6.2.1 [__call__](#)()

```
def skdaccess.framework.param_class.AutoList.__call__ (
    self ) [inherited]
```

Retrieve current list.

Returns

Current list

6.6.2.2 `__getitem__()`

```
def skdaccess.framework.param_class.AutoList.__getitem__ (
    self,
    ii ) [inherited]
```

Retrieves item from list.

Parameters

<i>ii</i>	Index of item to be retrieved
-----------	-------------------------------

Returns

Item at index ii

6.6.2.3 `__len__()`

```
def skdaccess.framework.param_class.AutoList.__len__ (
    self ) [inherited]
```

Retrieves the length of parameters contained in the list.

Returns

Number of elements in the list

6.6.2.4 `__setitem__()`

```
def skdaccess.framework.param_class.AutoList.__setitem__ (
    self,
    ii,
    val ) [inherited]
```

Set a value in the list.

Parameters

<i>ii</i>	Index of list to be set
<i>val</i>	Input value

6.6.2.5 `__str__()`

```
def skdaccess.framework.param_class.AutoList.__str__ (
    self ) [inherited]
```

String representation of class.

Returns

String containing all parameters in list

6.6.2.6 `getAllOptions()`

```
def skdaccess.framework.param_class.AutoList.getAllOptions (
    self ) [inherited]
```

Get all possible options.

Returns

List that contains every option that could possibly be selected

6.6.2.7 `perturb()`

```
def skdaccess.framework.param_class.AutoListSubset.perturb (
    self )
```

Perturb the list by selecting a random subset of the initial list.

6.6.2.8 `reset()`

```
def skdaccess.framework.param_class.AutoList.reset (
    self ) [inherited]
```

Reset current list to initial list.

6.6.2.9 val()

```
def skdaccess.framework.param_class.AutoList.val (
    self ) [inherited]
```

Retrieves current list of parameters.

Returns

List of current parameters

6.6.3 Member Data Documentation

6.6.3.1 val_init

```
skdaccess.framework.param_class.AutoList.val_init [inherited]
```

6.6.3.2 val_list

```
skdaccess.framework.param_class.AutoListSubset.val_list
```

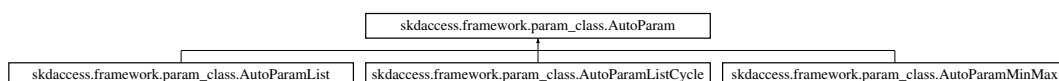
The documentation for this class was generated from the following file:

- [framework/param_class.py](#)

6.7 skdaccess.framework.param_class.AutoParam Class Reference

Defines a tunable parameter class inherited by specific subclasses.

Inheritance diagram for skdaccess.framework.param_class.AutoParam:



Public Member Functions

- def `__init__` (self, `val_init`)
Initialize an `AutoParam` object.
- def `perturb` (self)
Perturb paramter.
- def `reset` (self)
Reset value to initial value.
- def `__str__` (self)
String representation of class.
- def `__call__` (self)
Retrieves current value of the parameter.

Public Attributes

- `val`
- `val_init`

6.7.1 Detailed Description

Defines a tunable parameter class inherited by specific subclasses.

`AutoParam` class and subclass work on a single value. functions perturb value and reset to initial value

6.7.2 Constructor & Destructor Documentation

6.7.2.1 `__init__()`

```
def skdaccess.framework.param_class.AutoParam.__init__ (  
    self,  
    val_init )
```

Initialize an `AutoParam` object.

Parameters

<code>val_init</code>	Value for parameter
-----------------------	---------------------

6.7.3 Member Function Documentation

6.7.3.1 `__call__()`

```
def skdaccess.framework.param_class.AutoParam.__call__ (
    self )
```

Retrieves current value of the parameter.

Returns

Current value of the parameter

6.7.3.2 `__str__()`

```
def skdaccess.framework.param_class.AutoParam.__str__ (
    self )
```

String representation of class.

Returns

String of current value

6.7.3.3 `perturb()`

```
def skdaccess.framework.param_class.AutoParam.perturb (
    self )
```

Perturb paramter.

This class doesn't change the value.

6.7.3.4 `reset()`

```
def skdaccess.framework.param_class.AutoParam.reset (
    self )
```

Reset value to initial value.

6.7.4 Member Data Documentation

6.7.4.1 val

`skdaccess.framework.param_class.AutoParam.val`

6.7.4.2 val_init

`skdaccess.framework.param_class.AutoParam.val_init`

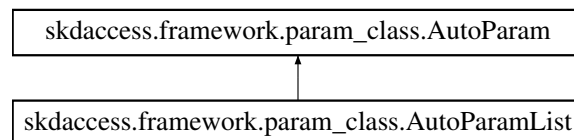
The documentation for this class was generated from the following file:

- [framework/param_class.py](#)

6.8 skdaccess.framework.param_class.AutoParamList Class Reference

A tunable parameter with a specified list of choices that can be randomly selected via perturb.

Inheritance diagram for `skdaccess.framework.param_class.AutoParamList`:



Public Member Functions

- `def __init__(self, val_init, val_list)`
Construct an [AutoParamList](#) object.
- `def perturb(self)`
Randomly select a value from `val_list`.
- `def reset(self)`
Reset the list to the default value.
- `def __str__(self)`
String representation of class.
- `def __call__(self)`
Retrieves current value of the parameter.

Public Attributes

- [val](#)
- [val_init](#)
- [val_list](#)

6.8.1 Detailed Description

A tunable parameter with a specified list of choices that can be randomly selected via perturb.

6.8.2 Constructor & Destructor Documentation

6.8.2.1 `__init__()`

```
def skdaccess.framework.param_class.AutoParamList.__init__ (
    self,
    val_init,
    val_list )
```

Construct an [AutoParamList](#) object.

Parameters

<i>val_init</i>	initial value for the parameter
<i>val_list</i>	List of possible variants for the parameter

6.8.3 Member Function Documentation

6.8.3.1 `__call__()`

```
def skdaccess.framework.param_class.AutoParam.__call__ (
    self ) [inherited]
```

Retrieves current value of the parameter.

Returns

Current value of the parameter

6.8.3.2 __str__()

```
def skdaccess.framework.param_class.AutoParam.__str__ (
    self ) [inherited]
```

String representation of class.

Returns

String of current value

6.8.3.3 perturb()

```
def skdaccess.framework.param_class.AutoParamList.perturb (
    self )
```

Randomly select a value from val_list.

6.8.3.4 reset()

```
def skdaccess.framework.param_class.AutoParamList.reset (
    self )
```

Reset the list to the default value.

6.8.4 Member Data Documentation

6.8.4.1 val

```
skdaccess.framework.param_class.AutoParamList.val
```

6.8.4.2 val_init

```
skdaccess.framework.param_class.AutoParamList.val_init
```

6.8.4.3 val_list

`skdaccess.framework.param_class.AutoParamList.val_list`

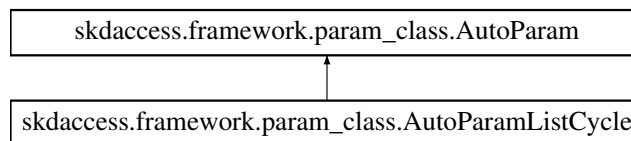
The documentation for this class was generated from the following file:

- [framework/param_class.py](#)

6.9 skdaccess.framework.param_class.AutoParamListCycle Class Reference

Cycles through a list of paramters.

Inheritance diagram for `skdaccess.framework.param_class.AutoParamListCycle`:



Public Member Functions

- `def __init__(self, val_list)`
Construct an [AutoParamListCycle](#).
- `def perturb(self)`
Select the next value from the list of parameters.
- `def reset(self)`
Reset the list to the default values.
- `def __str__(self)`
String representation of class.
- `def __call__(self)`
Retrieves current value of the parameter.

Public Attributes

- `val`
- `val_list`
- `current_index`
- `val_init`

6.9.1 Detailed Description

Cycles through a list of paramters.

6.9.2 Constructor & Destructor Documentation

6.9.2.1 __init__()

```
def skdaccess.framework.param_class.AutoParamListCycle.__init__ (
    self,
    val_list )
```

Construct an [AutoParamListCycle](#).

Parameters

<i>val_list</i>	List of possible variants for the parameter
-----------------	---

6.9.3 Member Function Documentation

6.9.3.1 __call__()

```
def skdaccess.framework.param_class.AutoParam.__call__ (
    self ) [inherited]
```

Retrieves current value of the parameter.

Returns

Current value of the parameter

6.9.3.2 __str__()

```
def skdaccess.framework.param_class.AutoParam.__str__ (
    self ) [inherited]
```

String representation of class.

Returns

String of current value

6.9.3.3 perturb()

```
def skdaccess.framework.param_class.AutoParamListCycle.perturb (
    self )
```

Select the next value from the list of parameters.

6.9.3.4 reset()

```
def skdaccess.framework.param_class.AutoParamListCycle.reset (
    self )
```

Reset the list to the default values.

6.9.4 Member Data Documentation

6.9.4.1 current_index

```
skdaccess.framework.param_class.AutoParamListCycle.current_index
```

6.9.4.2 val

```
skdaccess.framework.param_class.AutoParamListCycle.val
```

6.9.4.3 val_init

```
skdaccess.framework.param_class.AutoParam.val_init [inherited]
```

6.9.4.4 val_list

```
skdaccess.framework.param_class.AutoParamListCycle.val_list
```

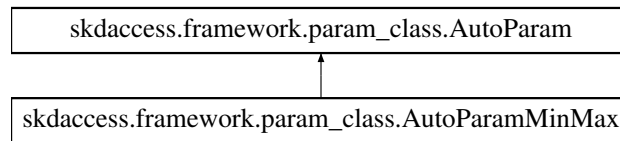
The documentation for this class was generated from the following file:

- framework/[param_class.py](#)

6.10 skdaccess.framework.param_class.AutoParamMinMax Class Reference

A tunable parameter with min and max ranges, perturbs to a random value in range.

Inheritance diagram for skdaccess.framework.param_class.AutoParamMinMax:



Public Member Functions

- def `__init__` (self, `val_init`, `val_min`, `val_max`, `decimals`=0, `extreme`=0)
Construct `AutoParamMinMax` object.
- def `perturb` (self)
Perturb the paramter by choosing a random value between `val_min` and `val_max`.
- def `reset` (self)
Reset to initial value.
- def `__str__` (self)
String representation of class.
- def `__call__` (self)
Retrieves current value of the parameter.

Public Attributes

- `val`
- `val_init`
- `val_min`
- `val_max`
- `n`
- `n_max`
- `decimals`

6.10.1 Detailed Description

A tunable parameter with min and max ranges, perturbs to a random value in range.

It can optionally choose either the min or the max after n perturbs

6.10.2 Constructor & Destructor Documentation

6.10.2.1 `__init__()`

```
def skdaccess.framework.param_class.AutoParamMinMax.__init__ (
    self,
    val_init,
    val_min,
    val_max,
    decimals = 0,
    extreme = 0 )
```

Construct [AutoParamMinMax](#) object.

Parameters

<i>val_init</i>	Initial value for parameter
<i>val_min</i>	Minimum value for param
<i>val_max</i>	Maximum value for parameter
<i>decimals</i>	Number of decimals to include in the random number
<i>extreme</i>	Either the maximum or minimum is chosen every extreme number of iterations. Using a value of one will be an extreme value every time. Using a value of zero will always choose a random value.

6.10.3 Member Function Documentation

6.10.3.1 `__call__()`

```
def skdaccess.framework.param_class.AutoParam.__call__ (
    self ) [inherited]
```

Retrieves current value of the parameter.

Returns

Current value of the parameter

6.10.3.2 `__str__()`

```
def skdaccess.framework.param_class.AutoParam.__str__ (
    self ) [inherited]
```

String representation of class.

Returns

String of current value

6.10.3.3 perturb()

```
def skdaccess.framework.param_class.AutoParamMinMax.perturb (
    self )
```

Perturb the parameter by choosing a random value between val_min and val_max.

Will choose a random number with precision specified by decimals. Will optionally pick the min or the max value after a specified number of perturb calls

6.10.3.4 reset()

```
def skdaccess.framework.param_class.AutoParamMinMax.reset (
    self )
```

Reset to initial value.

6.10.4 Member Data Documentation

6.10.4.1 decimals

```
skdaccess.framework.param_class.AutoParamMinMax.decimals
```

6.10.4.2 n

```
skdaccess.framework.param_class.AutoParamMinMax.n
```

6.10.4.3 n_max

```
skdaccess.framework.param_class.AutoParamMinMax.n_max
```

6.10.4.4 val

```
skdaccess.framework.param_class.AutoParamMinMax.val
```

6.10.4.5 val_init

```
skdaccess.framework.param_class.AutoParamMinMax.val_init
```

6.10.4.6 val_max

```
skdaccess.framework.param_class.AutoParamMinMax.val_max
```

6.10.4.7 val_min

```
skdaccess.framework.param_class.AutoParamMinMax.val_min
```

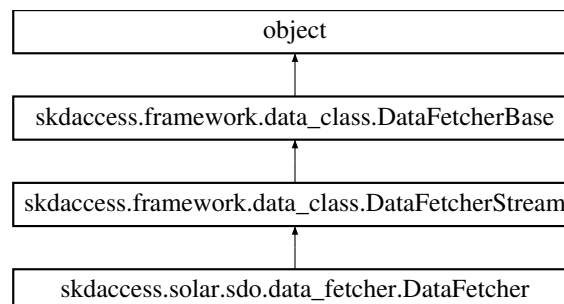
The documentation for this class was generated from the following file:

- [framework/param_class.py](#)

6.11 skdaccess.solar.sdo.DataFetcher Class Reference

Data Fetcher for the Solar Dynamics Observatory.

Inheritance diagram for skdaccess.solar.sdo.DataFetcher:



Public Member Functions

- def `__init__` (self, `ap_paramList`)
Initialize Solar Dynamics Observatory.
- def `output` (self)
Generate data wrapper.
- def `retrieveOnlineData` (self, `data_specification`)
Method for downloading data into memory.
- def `multirun_enabled` (self)
Returns whether or not this data fetcher is multirun enabled.
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- `ap_paramList`
- `verbose`

6.11.1 Detailed Description

Data Fetcher for the Solar Dynamics Observatory.

6.11.2 Constructor & Destructor Documentation

6.11.2.1 `__init__()`

```
def skdaccess.solar.sdo.DataFetcher.__init__ (
    self,
    ap_paramList )
```

Initialize Solar Dynamics Observatory.

Parameters

<i>ap_paramList[url_list]</i>	Autolist of URLs to access
-------------------------------	----------------------------

6.11.3 Member Function Documentation**6.11.3.1 __str__()**

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.11.3.2 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.11.3.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.11.3.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.11.3.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.11.3.6 output()

```
def skdaccess.solar.sdo.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

data wrapper of SDO data

6.11.3.7 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.11.3.8 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.11.3.9 retrieveOnlineData()

```
def skdaccess.framework.data_class.DataFetcherStream.retrieveOnlineData (
    self,
    data_specification ) [inherited]
```

Method for downloading data into memory.

Parameters

<i>data_specification</i>	Url list of data to be retrieved
---------------------------	----------------------------------

Returns

Retrieved data

6.11.3.10 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.11.3.11 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.11.3.12 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.11.4 Member Data Documentation

6.11.4.1 `ap_paramList`

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.11.4.2 `verbose`

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

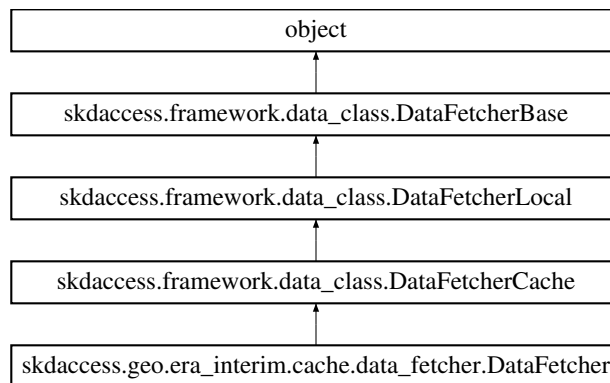
The documentation for this class was generated from the following file:

- [solar/sdo/data_fetcher.py](#)

6.12 `skdaccess.geo.era_interim.cache.DataFetcher` Class Reference

[DataFetcher](#) for retrieving ERA-I data.

Inheritance diagram for `skdaccess.geo.era_interim.cache.DataFetcher`:



Public Member Functions

- `def __init__ (self, date_list, data_names, username, password)`
- Initialize Data Fetcher.
- `def output (self)`
- Generate data wrapper.
- `def checkIfDataExists (self, in_file_name)`
- Checks if the file exists on the filesystem and the file is not empty.
- `def cacheData (self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)`
- Download and store specified data to local disk.
- `def multirun_enabled (self)`

- Returns whether or not this data fetcher is multirun enabled.*
- def [getHDFSStorage](#) (self, keyname)
Retrieve a Pandas HDF Store for a dataset.
- def [getDataLocation](#) (data_name)
Get the location of data set.
- def [setDataLocation](#) (data_name, location, key='data_location')
Set the location of a data set.
- def [perturb](#) (self)
Perturb parameters.
- def [reset](#) (self)
Set all parameters to initial value.
- def [__str__](#) (self)
Generate string description.
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [date_list](#)
- [data_names](#)
- [username](#)
- [password](#)
- [ap_paramList](#)
- [verbose](#)

6.12.1 Detailed Description

[DataFetcher](#) for retrieving ERA-I data.

6.12.2 Constructor & Destructor Documentation

6.12.2.1 `__init__()`

```
def skdaccess.geo.era_interim.cache.DataFetcher.__init__ (
    self,
    date_list,
    data_names,
    username,
    password )
```

Initialize Data Fetcher.

Parameters

<i>date_list</i>	list of dates
<i>data_names</i>	list of data names
<i>username</i>	UCAR username
<i>password</i>	UCAR password

6.12.3 Member Function Documentation

6.12.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.12.3.2 `cacheData()`

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.12.3.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.12.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.12.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.12.3.6 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.12.3.7 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.12.3.8 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.12.3.9 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.12.3.10 output()

```
def skdaccess.geo.era_interim.cache.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

Era-I weather in a data wrapper

6.12.3.11 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.12.3.12 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.12.3.13 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.12.3.14 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.12.3.15 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.12.3.16 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.12.4 Member Data Documentation

6.12.4.1 `ap_paramList`

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.12.4.2 `data_names`

`skdaccess.geo.era_interim.cache.DataFetcher.data_names`

6.12.4.3 `date_list`

`skdaccess.geo.era_interim.cache.DataFetcher.date_list`

6.12.4.4 `password`

`skdaccess.geo.era_interim.cache.DataFetcher.password`

6.12.4.5 `username`

`skdaccess.geo.era_interim.cache.DataFetcher.username`

6.12.4.6 `verbose`

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

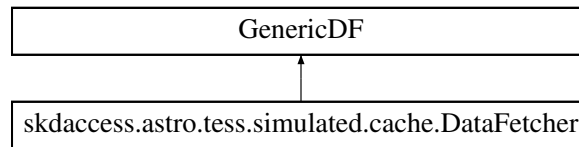
The documentation for this class was generated from the following file:

- `geo/era_interim/cache/data_fetcher.py`

6.13 skdaccess.astro.tess.simulated.cache.DataFetcher Class Reference

Data Fetcher for TESS data alerts.

Inheritance diagram for skdaccess.astro.tess.simulated.cache.DataFetcher:



Public Member Functions

- `def __init__ (self, ap_paramList)`
Initialize TESS Data Fetcher.
- `def generateURLFromTID (self, tid_list)`
Generate URL from TID.
- `def getTargetInformation ()`
Retrieve Target information for TESS data alerts.

Public Attributes

- `start_url`
- `end_url`

6.13.1 Detailed Description

Data Fetcher for TESS data alerts.

6.13.2 Constructor & Destructor Documentation

6.13.2.1 __init__()

```
def skdaccess.astro.tess.simulated.cache.DataFetcher.__init__ (
    self,
    ap_paramList )
```

Initialize TESS Data Fetcher.

Parameters

<i>ap_paramList[tess_ids]</i>	List of TESS IDs to retrieve
<i>start_url</i>	URL to prepend before the TESS ID
<i>end_url</i>	URL to append after the TESS ID

6.13.3 Member Function Documentation

6.13.3.1 generateURLFromTID()

```
def skdaccess.astro.tess.simulated.cache.DataFetcher.generateURLFromTID (
    self,
    tid_list )
```

Generate URL from TID.

Parameters

<i>tid_list</i>	Input Tess ID list
<i>return</i>	url to access data

6.13.3.2 getTargetInformation()

```
def skdaccess.astro.tess.simulated.cache.DataFetcher.getTargetInformation ( )
```

Retrieve Target information for TESS data alerts.

6.13.4 Member Data Documentation

6.13.4.1 end_url

```
skdaccess.astro.tess.simulated.cache.DataFetcher.end_url
```

6.13.4.2 start_url

```
skdaccess.astro.tess.simulated.cache.DataFetcher.start_url
```

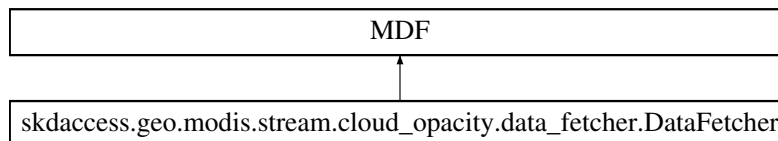
The documentation for this class was generated from the following file:

- astro/tess/simulated/[cache.py](#)

6.14 skdaccess.geo.modis.stream.cloud_opacity.DataFetcher Class Reference

Data Fetcher for MODIS Cloud Opacity.

Inheritance diagram for skdaccess.geo.modis.stream.cloud_opacity.DataFetcher:



Public Member Functions

- def `__init__` (self, ap_paramList, start_date, end_date, modis_platform='Terra', daynightboth='D', grid=None)
Construct Data Fetcher object for MODIS cloud Opacity data.

6.14.1 Detailed Description

Data Fetcher for MODIS Cloud Opacity.

6.14.2 Constructor & Destructor Documentation

6.14.2.1 __init__()

```
def skdaccess.geo.modis.stream.cloud_opacity.DataFetcher.__init__ (
    self,
    ap_paramList,
    start_date,
    end_date,
    modis_platform = 'Terra',
    daynightboth = 'D',
    grid = None )
```

Construct Data Fetcher object for MODIS cloud Opacity data.

Parameters

<i>ap_paramList[lat]</i>	Search latitude
<i>ap_paramList[lon]</i>	Search longitude
<i>start_date</i>	Starting date
<i>end_date</i>	Ending date
<i>modis_platform</i>	Platform (Either "Terra" or "Aqua")
<i>daynightboth</i>	Use daytime data ('D'), nighttime data ('N') or both ('B')
<i>grid</i>	Further divide each image into a multiple grids of size (y,x)

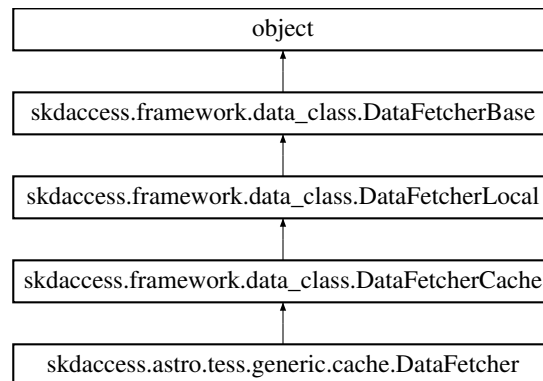
The documentation for this class was generated from the following file:

- [geo/modis/stream/cloud_opacity/data_fetcher.py](#)

6.15 skdaccess.astro.tess.generic.cache.DataFetcher Class Reference

Data Fetcher for TESS data alerts.

Inheritance diagram for skdaccess.astro.tess.generic.cache.DataFetcher:



Public Member Functions

- def `__init__` (self, [ap_paramList](#), [toi_information](#))
Initialize TESS Data Fetcher.
- def [getTargetInformation](#) ()
Retrieve Target list information.
- def [generateURLFromTID](#) (self, tid_list)
Generate URL from TID.
- def [output](#) (self)
Retrieve Tess data.
- def [checkIfDataExists](#) (self, in_file_name)
Checks if the file exists on the filesystem and the file is not empty.

- def [cacheData](#) (self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)
Download and store specified data to local disk.
- def [multirun_enabled](#) (self)
Returns whether or not this data fetcher is multirun enabled.
- def [getHDFSStorage](#) (self, keyname)
Retrieve a Pandas HDF Store for a dataset.
- def [getDataLocation](#) (data_name)
Get the location of data set.
- def [setDataLocation](#) (data_name, location, key='data_location')
Set the location of a data set.
- def [perturb](#) (self)
Perturb parameters.
- def [reset](#) (self)
Set all parameters to initial value.
- def [__str__](#) (self)
Generate string description.
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [toi_information](#)
- [ap_paramList](#)
- [verbose](#)

6.15.1 Detailed Description

Data Fetcher for TESS data alerts.

6.15.2 Constructor & Destructor Documentation

6.15.2.1 `__init__()`

```
def skdaccess.astro.tess.generic.cache.DataFetcher.__init__ (
    self,
    ap_paramList,
    toi_information )
```

Initialize TESS Data Fetcher.

Parameters

<i>ap_paramList[tess_ids]</i>	List of TESS IDs to retrieve
<i>toi_information</i>	Pandas dataframe containing target information

6.15.3 Member Function Documentation

6.15.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.15.3.2 `cacheData()`

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data

Parameters

<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.15.3.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.15.3.4 generateURLFromTID()

```
def skdaccess.astro.tess.generic.cache.DataFetcher.generateURLFromTID (
    self,
    tid_list )
```

Generate URL from TID.

Parameters

<i>tid_list</i>	Input Tess ID list
-----------------	--------------------

Returns

URL List of of objects in tid_list

6.15.3.5 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.15.3.6 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.15.3.7 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.15.3.8 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.15.3.9 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.15.3.10 `getTargetInformation()`

```
def skdaccess.astro.tess.generic.cache.DataFetcher.getTargetInformation ( )
```

Retrieve Target list information.

Returns

Target information list

6.15.3.11 `multirun_enabled()`

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.15.3.12 `output()`

```
def skdaccess.astro.tess.generic.cache.DataFetcher.output (
    self )
```

Retrieve Tess data.

Returns

TableWrapper containing TESS lightcurves

6.15.3.13 `perturb()`

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.15.3.14 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.15.3.15 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.15.3.16 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.15.3.17 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.15.3.18 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.15.4 Member Data Documentation

6.15.4.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]
```

6.15.4.2 toi_information

```
skdaccess.astro.tess.generic.cache.DataFetcher.toi_information
```


6.15.4.3 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

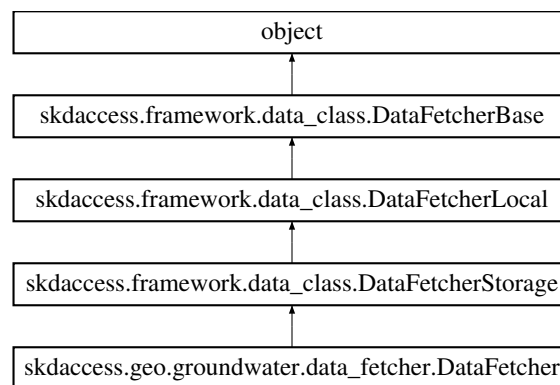
The documentation for this class was generated from the following file:

- [astro/tess/generic/cache.py](#)

6.16 skdaccess.geo.groundwater.DataFetcher Class Reference

Generates Data Wrappers of groundwater measurements taken in the US.

Inheritance diagram for `skdaccess.geo.groundwater.DataFetcher`:



Public Member Functions

- `def __init__(self, ap_paramList=[], start_date=None, end_date=None, cutoff=0.75)`
Construct a Groundwater Data Fetcher.
- `def output(self)`
Fetch Groundwater Data Wrapper.
- `def __str__(self)`
String representation of data fetcher.
- `def getStationMetadata()`
Retrieve metadata on groundwater wells.
- `def downloadFullDataset(cls, out_file='gw.h5', use_file=None)`
Download and parse US groundwater data provided by USGS.
- `def multirun_enabled(self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getDataLocation(data_name)`
Get the location of data set.
- `def setDataLocation(data_name, location, key='data_location')`
Set the location of a data set.
- `def perturb(self)`

- Perturb parameters.*
 - def `reset` (self)
 - Set all parameters to initial value.*
 - def `getMetadata` (self)
 - Return metadata about Data Fetcher.*
 - def `getConfig` ()
 - Retrieve skdaccess configuration.*
 - def `getConfigItem` (section, key)
 - Retrieve skdaccess configuration item.*
 - def `writeConfigItem` (section, key, value)
 - Retrieve skdaccess configuration item.*
 - def `writeConfig` (conf)
 - Write config to disk.*
 - def `verbose_print` (self, args, kwargs)
 - Print statement if verbose flag is set.*

Public Attributes

- `start_date`
- `end_date`
- `ap_paramList`
- `cutoff`
- `verbose`

6.16.1 Detailed Description

Generates Data Wrappers of groundwater measurements taken in the US.

6.16.2 Constructor & Destructor Documentation

6.16.2.1 `__init__()`

```
def skdaccess.geo.groundwater.DataFetcher.__init__ (
    self,
    ap_paramList = [],
    start_date = None,
    end_date = None,
    cutoff = 0.75 )
```

Construct a Groundwater Data Fetcher.

Parameters

<i>ap_paramList[LowerLat]</i>	Autoparam Lower latitude
<i>ap_paramList[UpperLat]</i>	Autoparam Upper latitude
<i>ap_paramList[LeftLon]</i>	Autoparam Left longitude
<i>ap_paramList[RightLon]</i>	Autoparam Right longitude
<i>start_date</i>	Starting date (default: None)
<i>end_date</i>	Ending date (default: None)
<i>cutoff</i>	Required amount of data for each station

6.16.3 Member Function Documentation

6.16.3.1 `__str__()`

```
def skdaccess.geo.groundwater.DataFetcher.__str__ (
    self )
```

String representation of data fetcher.

Returns

string describing data fetcher

6.16.3.2 `downloadFullDataset()`

```
def skdaccess.geo.groundwater.DataFetcher.downloadFullDataset (
    cls,
    out_file = 'gw.h5',
    use_file = None )
```

Download and parse US groundwater data provided by USGS.

Parameters

<i>out_file</i>	Output filename for parsed data
<i>use_file</i>	Specify the directory where the data is. If None, the function will download the data

Returns

Absolute path of parsed data

6.16.3.3 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.16.3.4 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.16.3.5 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.16.3.6 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.16.3.7 getStationMetadata()

```
def skdaccess.geo.groundwater.DataFetcher.getStationMetadata ( )
```

Retrieve metadata on groundwater wells.

Returns

pandas dataframe with groundwater well information

6.16.3.8 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStorage.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.16.3.9 output()

```
def skdaccess.geo.groundwater.DataFetcher.output (
    self )
```

Fetch Groundwater Data Wrapper.

Returns

Groundwater Data Wrapper

6.16.3.10 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.16.3.11 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.16.3.12 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.16.3.13 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.16.3.14 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.16.3.15 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.16.4 Member Data Documentation**6.16.4.1 ap_paramList**

`skdaccess.geo.groundwater.DataFetcher.ap_paramList`

6.16.4.2 cutoff

`skdaccess.geo.groundwater.DataFetcher.cutoff`

6.16.4.3 end_date

`skdaccess.geo.groundwater.DataFetcher.end_date`

6.16.4.4 start_date

`skdaccess.geo.groundwater.DataFetcher.start_date`

6.16.4.5 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

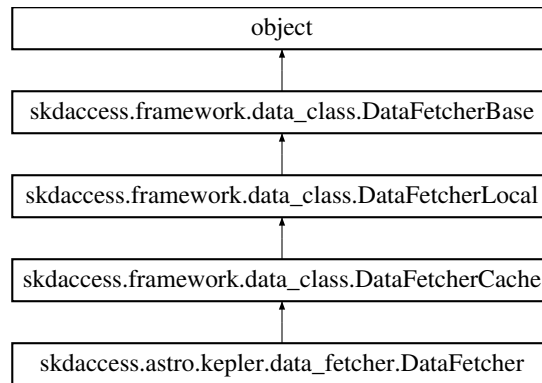
The documentation for this class was generated from the following file:

- [geo/groundwater/data_fetcher.py](#)

6.17 skdaccess.astro.kepler.DataFetcher Class Reference

Data Fetcher for Kepler light curve data.

Inheritance diagram for skdaccess.astro.kepler.DataFetcher:



Public Member Functions

- def `__init__` (self, `ap_paramList`, `quarter_list`=None)
Initialize Kepler Data Fetcher.
- def `downloadKeplerData` (self, `kid_list`)
Download and parse Kepler data for a list of kepler id's.
- def `cacheData` (self, `data_specification`)
Cache Kepler data locally.
- def `output` (self)
Output kepler data wrapper.
- def `checkIfDataExists` (self, `in_file_name`)
Checks if the file exists on the filesystem and the file is not empty.
- def `cacheData` (self, `keyname`, `online_path_list`, `username`=None, `password`=None, `authentication_url`=None, `cookiejar`=None, `use_requests`=False, `use_progress_bar`=True)
Download and store specified data to local disk.
- def `multirun_enabled` (self)
Returns whether or not this data fetcher is multirun enabled.
- def `getHDFStorage` (self, `keyname`)
Retrieve a Pandas HDF Store for a dataset.
- def `getDataLocation` (`data_name`)
Get the location of data set.
- def `setDataLocation` (`data_name`, `location`, `key`='data_location')
Set the location of a data set.
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)

- *Generate string description.*
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [quarter_list](#)
- [ap_paramList](#)
- [verbose](#)

6.17.1 Detailed Description

Data Fetcher for Kepler light curve data.

6.17.2 Constructor & Destructor Documentation

6.17.2.1 `__init__()`

```
def skdaccess.astro.kepler.DataFetcher.__init__ (
    self,
    ap_paramList,
    quarter_list = None )
```

Initialize Kepler Data Fetcher.

Parameters

<code>ap_paramList[kepler_id_list]</code>	List of kepler id's
<code>quarter_list</code>	List of quarters (0-17) (default: all quarters)

6.17.3 Member Function Documentation

6.17.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.17.3.2 `cacheData()` [1/2]

```
def skdaccess.astro.kepler.DataFetcher.cacheData (
    self,
    data_specification )
```

Cache Kepler data locally.

Parameters

<i>data_specification</i>	List of kepler IDs
---------------------------	--------------------

6.17.3.3 `cacheData()` [2/2]

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data

Parameters

<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.17.3.4 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.17.3.5 downloadKeplerData()

```
def skdaccess.astro.kepler.DataFetcher.downloadKeplerData (
    self,
    kid_list )
```

Download and parse Kepler data for a list of kepler id's.

Parameters

<i>kid_list</i>	List of Kepler ID's to download
-----------------	---------------------------------

Returns

dictionary of kepler data

6.17.3.6 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.17.3.7 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.17.3.8 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.17.3.9 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.17.3.10 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.17.3.11 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.17.3.12 output()

```
def skdaccess.astro.kepler.DataFetcher.output (
    self )
```

Output kepler data wrapper.

Returns

DataWrapper

6.17.3.13 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.17.3.14 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.17.3.15 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.17.3.16 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.17.3.17 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.17.3.18 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.17.4 Member Data Documentation**6.17.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.17.4.2 quarter_list

`skdaccess.astro.kepler.DataFetcher.quarter_list`

6.17.4.3 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

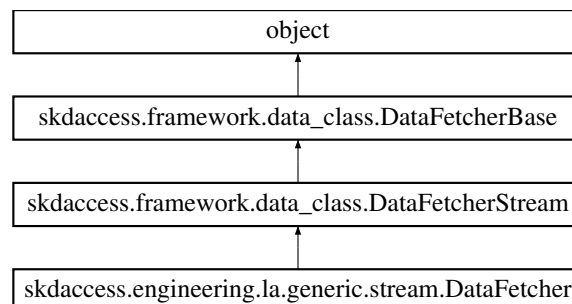
The documentation for this class was generated from the following file:

- [astro/kepler/data_fetcher.py](#)

6.18 skdaccess.engineering.la.generic.stream.DataFetcher Class Reference

Class for handling data requests to data.lacity.org.

Inheritance diagram for `skdaccess.engineering.la.generic.stream.DataFetcher`:



Public Member Functions

- `def __init__ (self, endpoint, parameters, label, verbose=False, app_token=None, pandas_kwargs)`
Initialize Data Fetcher for accessing data.lacity.org.
- `def output (self)`
Retrieve data from data.lacity.org.
- `def retrieveOnlineData (self, data_specification)`
Method for downloading data into memory.
- `def multirun_enabled (self)`
Returns whether or not this data fetcher is multirun enabled.
- `def perturb (self)`
Perturb parameters.
- `def reset (self)`
Set all parameters to initial value.
- `def __str__ (self)`
Generate string description.
- `def getMetadata (self)`
Return metadata about Data Fetcher.
- `def getConfig ()`
Retrieve skdaccess configuration.
- `def getConfigItem (section, key)`
Retrieve skdaccess configuration item.
- `def writeConfigItem (section, key, value)`
Retrieve skdaccess configuration item.
- `def writeConfig (conf)`
Write config to disk.
- `def verbose_print (self, args, kwargs)`
Print statement if verbose flag is set.

Public Attributes

- [base_url](#)
- [base_url_and_endpoint](#)
- [parameters](#)
- [label](#)
- [app_token](#)
- [pandas_kwargs](#)
- [ap_paramList](#)
- [verbose](#)

6.18.1 Detailed Description

Class for handling data requests to data.lacity.org.

6.18.2 Constructor & Destructor Documentation

6.18.2.1 `__init__()`

```
def skdaccess.engineering.la.generic.stream.DataFetcher.__init__ (
    self,
    endpoint,
    parameters,
    label,
    verbose = False,
    app_token = None,
    pandas_kwargs )
```

Initialize Data Fetcher for accessing data.lacity.org.

Parameters

<i>endpoint</i>	Data endpoint string
<i>parameters</i>	Parameters to use when retrieving dta
<i>label</i>	Label of pandas dataframe
<i>verbose</i>	Print out extra information
<i>app_token</i>	Application token to use to avoid throttling issues
<i>date_columns</i>	
<i>pandas_kwargs</i>	Any additional key word arguments are passed to pandas.read_csv

6.18.3 Member Function Documentation

6.18.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.18.3.2 `getConfig()`

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.18.3.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.18.3.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.18.3.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.18.3.6 output()

```
def skdaccess.engineering.la.generic.stream.DataFetcher.output (
    self )
```

Retrieve data from data.lacity.org.

Returns

Table wrapper of containing specified data

6.18.3.7 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.18.3.8 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.18.3.9 retrieveOnlineData()

```
def skdaccess.framework.data_class.DataFetcherStream.retrieveOnlineData (
    self,
    data_specification ) [inherited]
```

Method for downloading data into memory.

Parameters

<i>data_specification</i>	Url list of data to be retrieved
---------------------------	----------------------------------

Returns

Retrieved data

6.18.3.10 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.18.3.11 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.18.3.12 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.18.4 Member Data Documentation**6.18.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.18.4.2 app_token

`skdaccess.engineering.la.generic.stream.DataFetcher.app_token`

6.18.4.3 base_url

`skdaccess.engineering.la.generic.stream.DataFetcher.base_url`

6.18.4.4 base_url_and_endpoint

`skdaccess.engineering.la.generic.stream.DataFetcher.base_url_and_endpoint`

6.18.4.5 label

`skdaccess.engineering.la.generic.stream.DataFetcher.label`

6.18.4.6 pandas_kwargs

`skdaccess.engineering.la.generic.stream.DataFetcher.pandas_kwargs`

6.18.4.7 parameters

`skdaccess.engineering.la.generic.stream.DataFetcher.parameters`

6.18.4.8 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

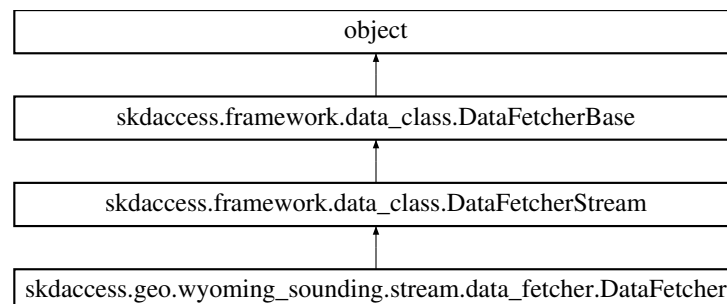
The documentation for this class was generated from the following file:

- [engineering/la/generic/stream.py](#)

6.19 skdaccess.geo.wyoming_sounding.stream.DataFetcher Class Reference

[DataFetcher](#) for retrieving Wyoming Sounding data.

Inheritance diagram for `skdaccess.geo.wyoming_sounding.stream.DataFetcher`:



Public Member Functions

- def `__init__` (self, `station_number`, year, month, `day_start`, `day_end`, `start_hour`=0, `end_hour`=12)
Initialize Data Fetcher.
- def `output` (self, `shared_lock`=None, `shared_list`=None)
Generate data wrapper.
- def `retrieveOnlineData` (self, `data_specification`)
Method for downloading data into memory.
- def `multirun_enabled` (self)
Returns whether or not this data fetcher is multirun enabled.
- def `output` (self)
Output data wrapper.
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- `station_number`
- `year_list`
- `month_list`
- `day_start`
- `day_end`
- `start_hour`
- `end_hour`
- `ap_paramList`
- `verbose`

6.19.1 Detailed Description

[DataFetcher](#) for retrieving Wyoming Sounding data.

6.19.2 Constructor & Destructor Documentation

6.19.2.1 `__init__()`

```
def skdaccess.geo.wyoming_sounding.stream.DataFetcher.__init__ (
    self,
    station_number,
    year,
    month,
    day_start,
    day_end,
    start_hour = 0,
    end_hour = 12 )
```

Initialize Data Fetcher.

Parameters

<i>station_number</i>	Station number
<i>year</i>	Input year
<i>month</i>	Input month (Integer for a single month, or a list of integers for multiple months)
<i>day_start</i>	First day of the month to include
<i>day_end</i>	Last day of the month to include
<i>start_hour</i>	Starting hour (may be either 0 or 12)
<i>end_hour</i>	Ending hour (may be either 0 or 12)

6.19.3 Member Function Documentation

6.19.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.19.3.2 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.19.3.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.19.3.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.19.3.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.19.3.6 output() [1/2]

```
def skdaccess.framework.data_class.DataFetcherBase.output (
    self ) [inherited]
```

Output data wrapper.

Returns

Datawrapper

6.19.3.7 output() [2/2]

```
def skdaccess.geo.wyoming_sounding.stream.DataFetcher.output (
    self,
    shared_lock = None,
    shared_list = None )
```

Generate data wrapper.

Returns

Wyoming sounding data in a data wrapper

6.19.3.8 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.19.3.9 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.19.3.10 retrieveOnlineData()

```
def skdaccess.framework.data_class.DataFetcherStream.retrieveOnlineData (
    self,
    data_specification ) [inherited]
```

Method for downloading data into memory.

Parameters

<i>data_specification</i>	Url list of data to be retrieved
---------------------------	----------------------------------

Returns

Retrieved data

6.19.3.11 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.19.3.12 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
```

```
conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.19.3.13 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (  
    section,  
    key,  
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.19.4 Member Data Documentation

6.19.4.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]
```

6.19.4.2 day_end

```
skdaccess.geo.wyoming_sounding.stream.DataFetcher.day_end
```

6.19.4.3 day_start

`skdaccess.geo.wyoming_sounding.stream.DataFetcher.day_start`

6.19.4.4 end_hour

`skdaccess.geo.wyoming_sounding.stream.DataFetcher.end_hour`

6.19.4.5 month_list

`skdaccess.geo.wyoming_sounding.stream.DataFetcher.month_list`

6.19.4.6 start_hour

`skdaccess.geo.wyoming_sounding.stream.DataFetcher.start_hour`

6.19.4.7 station_number

`skdaccess.geo.wyoming_sounding.stream.DataFetcher.station_number`

6.19.4.8 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

6.19.4.9 year_list

`skdaccess.geo.wyoming_sounding.stream.DataFetcher.year_list`

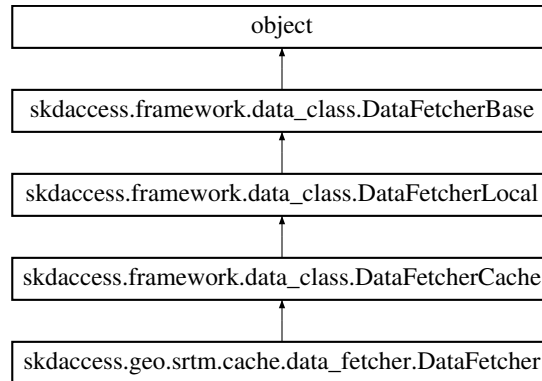
The documentation for this class was generated from the following file:

- `geo/wyoming_sounding/stream/data_fetcher.py`

6.20 skdaccess.geo.srtm.cache.DataFetcher Class Reference

[DataFetcher](#) for retrieving data from the Shuttle Radar Topography Mission.

Inheritance diagram for skdaccess.geo.srtm.cache.DataFetcher:



Public Member Functions

- `def __init__ (self, lat_tile_start, lat_tile_end, lon_tile_start, lon_tile_end, username, password, arcsecond_↵ sampling=1, mask_water=True, store_geolocation_grids=False)`
Initialize Data Fetcher.
- `def output (self)`
Generate SRTM data wrapper.
- `def checkIfDataExists (self, in_file_name)`
Checks if the file exists on the filesystem and the file is not empty.
- `def cacheData (self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)`
Download and store specified data to local disk.
- `def multirun_enabled (self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getHDFStorage (self, keyname)`
Retrieve a Pandas HDF Store for a dataset.
- `def getDataLocation (data_name)`
Get the location of data set.
- `def setDataLocation (data_name, location, key='data_location')`
Set the location of a data set.
- `def perturb (self)`
Perturb parameters.
- `def reset (self)`
Set all parameters to initial value.
- `def __str__ (self)`
Generate string description.
- `def getMetadata (self)`
Return metadata about Data Fetcher.

- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [lat_tile_start](#)
- [lat_tile_end](#)
- [lon_tile_start](#)
- [lon_tile_end](#)
- [username](#)
- [password](#)
- [arcsecond_sampling](#)
Determine the longitude and latitude of the lowerleft corner of the input filename.
- [mask_water](#)
- [store_geolocation_grids](#)
- [ap_paramList](#)
- [verbose](#)

6.20.1 Detailed Description

[DataFetcher](#) for retrieving data from the Shuttle Radar Topography Mission.

6.20.2 Constructor & Destructor Documentation

6.20.2.1 `__init__()`

```
def skdaccess.geo.srtm.cache.DataFetcher.__init__ (
    self,
    lat_tile_start,
    lat_tile_end,
    lon_tile_start,
    lon_tile_end,
    username,
    password,
    arcsecond_sampling = 1,
    mask_water = True,
    store_geolocation_grids = False )
```

Initialize Data Fetcher.

Parameters

<i>lat_tile_start</i>	Latitude of the southwest corner of the starting tile
<i>lat_tile_end</i>	Latitude of the southwest corner of the last tile
<i>lon_tile_start</i>	Longitude of the southwest corner of the starting tile
<i>lon_tile_end</i>	Longitude of the southwest corner of the last tile
<i>username</i>	NASA Earth Data username
<i>password</i>	NASA Earth Data Password
<i>arcsecond_sampling</i>	Sample spacing of the SRTM data, either 1 arc- second or 3 arc-seconds
<i>mask_water</i>	True if the water bodies should be masked, false otherwise
<i>store_geolocation_grids</i>	Store grids of latitude and longitude in the metadata

6.20.3 Member Function Documentation

6.20.3.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.20.3.2 cacheData()

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use progress bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.20.3.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.20.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.20.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.20.3.6 `getDataLocation()`

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.20.3.7 `getHDFStorage()`

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.20.3.8 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.20.3.9 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.20.3.10 output()

```
def skdaccess.geo.srtm.cache.DataFetcher.output (
    self )
```

Generate SRTM data wrapper.

Returns

SRTM Image Wrapper

6.20.3.11 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.20.3.12 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.20.3.13 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.20.3.14 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.20.3.15 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.20.3.16 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.20.4 Member Data Documentation

6.20.4.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]
```

6.20.4.2 arcsecond_sampling

```
skdaccess.geo.srtm.cache.DataFetcher.arcsecond_sampling
```

Determine the longitude and latitude of the lowerleft corner of the input filename.

Parameters

<i>in_filename</i>	Input SRTM filename
--------------------	---------------------

Returns

Latitude of southwest corner, Longitude of southwest corner

6.20.4.3 lat_tile_end

```
skdaccess.geo.srtm.cache.DataFetcher.lat_tile_end
```

6.20.4.4 lat_tile_start

```
skdaccess.geo.srtm.cache.DataFetcher.lat_tile_start
```

6.20.4.5 lon_tile_end

```
skdaccess.geo.srtm.cache.DataFetcher.lon_tile_end
```

6.20.4.6 lon_tile_start

```
skdaccess.geo.srtm.cache.DataFetcher.lon_tile_start
```

6.20.4.7 mask_water

```
skdaccess.geo.srtm.cache.DataFetcher.mask_water
```


6.20.4.8 password

```
skdaccess.geo.srtm.cache.DataFetcher.password
```

6.20.4.9 store_geolocation_grids

```
skdaccess.geo.srtm.cache.DataFetcher.store_geolocation_grids
```

6.20.4.10 username

```
skdaccess.geo.srtm.cache.DataFetcher.username
```

6.20.4.11 verbose

```
skdaccess.framework.data_class.DataFetcherBase.verbose [inherited]
```

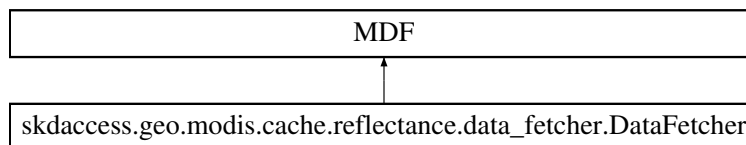
The documentation for this class was generated from the following file:

- [geo/srtm/cache/data_fetcher.py](#)

6.21 skdaccess.geo.modis.cache.reflectance.DataFetcher Class Reference

Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

Inheritance diagram for skdaccess.geo.modis.cache.reflectance.DataFetcher:



Public Member Functions

- `def __init__ (self, ap_paramList, start_date, end_date, modis_platform='Terra', daynightboth='D', grid=None, bands=[1])`

Construct Data Fetcher for MODIS 1km surface reflectance.

6.21.1 Detailed Description

Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

6.21.2 Constructor & Destructor Documentation

6.21.2.1 `__init__()`

```
def skdaccess.geo.modis.cache.reflectance.DataFetcher.__init__ (
    self,
    ap_paramList,
    start_date,
    end_date,
    modis_platform = 'Terra',
    daynightboth = 'D',
    grid = None,
    bands = [1 ]
```

Construct Data Fetcher for MODIS 1km surface reflectance.

Parameters

<i>ap_paramList[lat]</i>	Search latitude
<i>ap_paramList[lon]</i>	Search longitude
<i>start_date</i>	Starting date
<i>end_date</i>	Ending date
<i>modis_platform</i>	Platform (Either "Terra" or "Aqua")
<i>daynightboth</i>	Use daytime data ('D'), nighttime data ('N') or both ('B')
<i>grid</i>	Further divide each image into a multiple grids of size (y,x)
<i>bands</i>	List of modis bands to retrieve

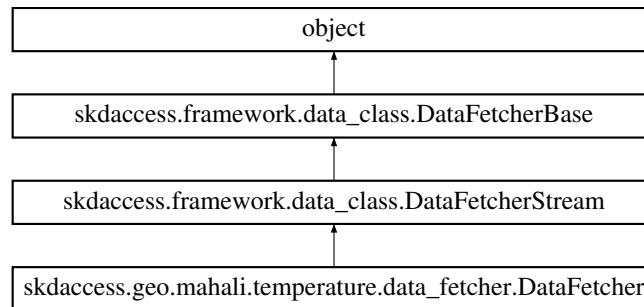
The documentation for this class was generated from the following file:

- [geo/modis/cache/reflectance/data_fetcher.py](#)

6.22 skdaccess.geo.mahali.temperature.DataFetcher Class Reference

Data Fetcher for Mahali temperature data.

Inheritance diagram for skdaccess.geo.mahali.temperature.DataFetcher:



Public Member Functions

- `def __init__ (self, ap_paramList=[], start_date=None, end_date=None)`
Initialize Mahali temperature data fetcher.
- `def retrieveOnlineData (self, data_specification)`
Load data in from a remote source.
- `def output (self)`
Generate data wrapper for Mahali temperatures.
- `def multirun_enabled (self)`
Returns whether or not this data fetcher is multirun enabled.
- `def perturb (self)`
Perturb parameters.
- `def reset (self)`
Set all parameters to initial value.
- `def __str__ (self)`
Generate string description.
- `def getMetadata (self)`
Return metadata about Data Fetcher.
- `def getConfig ()`
Retrieve skdaccess configuration.
- `def getConfigItem (section, key)`
Retrieve skdaccess configuration item.
- `def writeConfigItem (section, key, value)`
Retrieve skdaccess configuration item.
- `def writeConfig (conf)`
Write config to disk.
- `def verbose_print (self, args, kwargs)`
Print statement if verbose flag is set.

Public Attributes

- `start_date`
- `end_date`
- `ap_paramList`
- `verbose`

6.22.1 Detailed Description

Data Fetcher for Mahali temperature data.

6.22.2 Constructor & Destructor Documentation

6.22.2.1 `__init__()`

```
def skdaccess.geo.mahali.temperature.DataFetcher.__init__ (
    self,
    ap_paramList = [],
    start_date = None,
    end_date = None )
```

Initialize Mahali temperature data fetcher.

Parameters

<i>ap_paramList</i> [stations]	Autolist of stations (Defaults to all stations)
<i>start_date</i>	Starting date for seelcting data (Defaults to beginning of available data)
<i>end_date</i>	Ending date for selecting data (Defaults to end of available data)

6.22.3 Member Function Documentation

6.22.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.22.3.2 `getConfig()`

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.22.3.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.22.3.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.22.3.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.22.3.6 output()

```
def skdaccess.geo.mahali.temperature.DataFetcher.output (
    self )
```

Generate data wrapper for Mahali temperatures.

Returns

Mahali temperature data wrapper

6.22.3.7 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.22.3.8 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.22.3.9 retrieveOnlineData()

```
def skdaccess.geo.mahali.temperature.DataFetcher.retrieveOnlineData (
    self,
    data_specification )
```

Load data in from a remote source.

Parameters

<i>data_specification</i>	Pandas dataframe containing the columns 'station', 'date', and 'filename'
---------------------------	---

Returns

Ordered dictionary for each station (key) which contains a pandas data frame of the temperature

6.22.3.10 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.22.3.11 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.22.3.12 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.22.4 Member Data Documentation**6.22.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.22.4.2 end_date

`skdaccess.geo.mahali.temperature.DataFetcher.end_date`

6.22.4.3 start_date

`skdaccess.geo.mahali.temperature.DataFetcher.start_date`

6.22.4.4 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

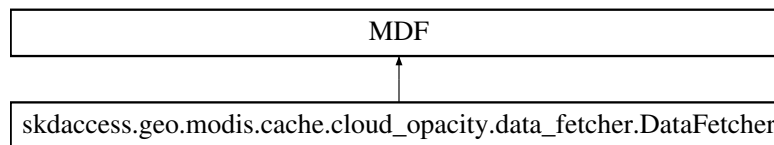
The documentation for this class was generated from the following file:

- `geo/mahali/temperature/data_fetcher.py`

6.23 skdaccess.geo.modis.cache.cloud_opacity.DataFetcher Class Reference

Data Fetcher for MODIS Cloud Opacity.

Inheritance diagram for skdaccess.geo.modis.cache.cloud_opacity.DataFetcher:



Public Member Functions

- def `__init__` (self, ap_paramList, start_date, end_date, modis_platform='Terra', daynightboth='D', grid=None)
Construct Data Fetcher object for MODIS cloud Opacity data.

6.23.1 Detailed Description

Data Fetcher for MODIS Cloud Opacity.

6.23.2 Constructor & Destructor Documentation

6.23.2.1 `__init__()`

```

def skdaccess.geo.modis.cache.cloud_opacity.DataFetcher.__init__ (
    self,
    ap_paramList,
    start_date,
    end_date,
    modis_platform = 'Terra',
    daynightboth = 'D',
    grid = None )

```

Construct Data Fetcher object for MODIS cloud Opacity data.

Parameters

<code>ap_paramList[lat]</code>	Search latitude
<code>ap_paramList[lon]</code>	Search longitude
<code>start_date</code>	Starting date
<code>end_date</code>	Ending date
<code>modis_platform</code>	Platform (Either "Terra" or "Aqua")
<code>daynightboth</code>	Use daytime data ('D'), nighttime data ('N') or both ('B')
<code>grid</code>	Further divide each image into a multiple grids of size (y,x)

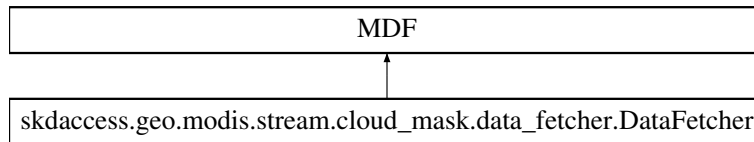
The documentation for this class was generated from the following file:

- [geo/modis/cache/cloud_opacity/data_fetcher.py](#)

6.24 skdaccess.geo.modis.stream.cloud_mask.DataFetcher Class Reference

Data Fetcher for MODIS Cloud Mask.

Inheritance diagram for skdaccess.geo.modis.stream.cloud_mask.DataFetcher:



Public Member Functions

- `def __init__(self, ap_paramList, start_date, end_date, modis_platform='Terra', daynightboth='D', grid=None)`
Construct Data Fetcher for MODIS cloud mask data.

6.24.1 Detailed Description

Data Fetcher for MODIS Cloud Mask.

6.24.2 Constructor & Destructor Documentation

6.24.2.1 __init__()

```

def skdaccess.geo.modis.stream.cloud_mask.DataFetcher.__init__(
    self,
    ap_paramList,
    start_date,
    end_date,
    modis_platform = 'Terra',
    daynightboth = 'D',
    grid = None )
  
```

Construct Data Fetcher for MODIS cloud mask data.

Parameters

<i>ap_paramList[lat]</i>	Search latitude
<i>ap_paramList[lon]</i>	Search longitude
<i>start_date</i>	Starting date
<i>end_date</i>	Ending date
<i>modis_platform</i>	Platform (Either "Terra" or "Aqua")
<i>daynightboth</i>	Use daytime data ('D'), nighttime data ('N') or both ('B')
<i>grid</i>	Further divide each image into a multiple grids of size (y,x)

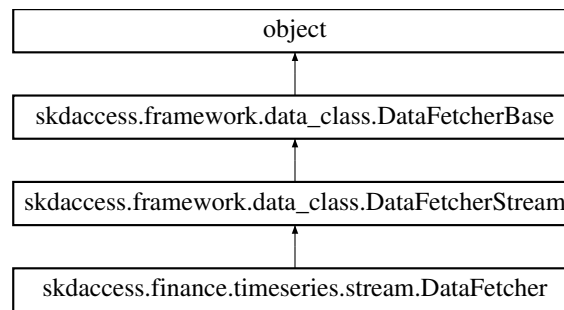
The documentation for this class was generated from the following file:

- [geo/modis/stream/cloud_mask/data_fetcher.py](#)

6.25 skdaccess.finance.timeseries.stream.DataFetcher Class Reference

Data Fetcher for retrieving stock data.

Inheritance diagram for skdaccess.finance.timeseries.stream.DataFetcher:



Public Member Functions

- `def __init__(self, ap_paramList, data_type, start_date=None, end_date=None, interval=None)`
- `def output(self)`
Retrieve stock data.
- `def retrieveOnlineData(self, data_specification)`
Method for downloading data into memory.
- `def multirun_enabled(self)`
Returns whether or not this data fetcher is multirun enabled.
- `def perturb(self)`
Perturb parameters.
- `def reset(self)`
Set all parameters to initial value.
- `def __str__(self)`

- *Generate string description.*
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [data_type](#)
- [start_date](#)
- [end_date](#)
- [interval](#)
- [possible_intervals](#)
- [possible_data_types](#)
- [ap_paramList](#)
- [verbose](#)

6.25.1 Detailed Description

Data Fetcher for retrieving stock data.

6.25.2 Constructor & Destructor Documentation

6.25.2.1 `__init__()`

```
def skdaccess.finance.timeseries.stream.DataFetcher.__init__ (
    self,
    ap_paramList,
    data_type,
    start_date = None,
    end_date = None,
    interval = None )
```

Parameters

<i>ap_paramList</i> [<i>stock_symbol_list</i>]	AutoList of stock symbols
<i>data_type</i>	Type of data to retrieve (daily, daily_adjusted, intraday, monthly, monthly_adjusted, weekly, weekly_adjusted)
<i>start_date</i>	Starting date
<i>end_date</i>	Ending date
<i>interval</i>	Interval for intraday (1min, 5min, 15min, 30min, 60min)

Returns

: Table data wrapper of stock data

6.25.3 Member Function Documentation

6.25.3.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.25.3.2 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.25.3.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.25.3.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.25.3.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.25.3.6 output()

```
def skdaccess.finance.timeseries.stream.DataFetcher.output (
    self )
```

Retrieve stock data.

Returns

TableWrapper of stock data

6.25.3.7 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.25.3.8 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.25.3.9 retrieveOnlineData()

```
def skdaccess.framework.data_class.DataFetcherStream.retrieveOnlineData (
    self,
    data_specification ) [inherited]
```

Method for downloading data into memory.

Parameters

<i>data_specification</i>	Url list of data to be retrieved
---------------------------	----------------------------------

Returns

Retrieved data

6.25.3.10 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.25.3.11 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.25.3.12 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.25.4 Member Data Documentation

6.25.4.1 ap_paramList

skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]

6.25.4.2 data_type

skdaccess.finance.timeseries.stream.DataFetcher.data_type

6.25.4.3 end_date

skdaccess.finance.timeseries.stream.DataFetcher.end_date

6.25.4.4 interval

skdaccess.finance.timeseries.stream.DataFetcher.interval

6.25.4.5 possible_data_types

skdaccess.finance.timeseries.stream.DataFetcher.possible_data_types

6.25.4.6 possible_intervals

skdaccess.finance.timeseries.stream.DataFetcher.possible_intervals

6.25.4.7 start_date

skdaccess.finance.timeseries.stream.DataFetcher.start_date

6.25.4.8 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

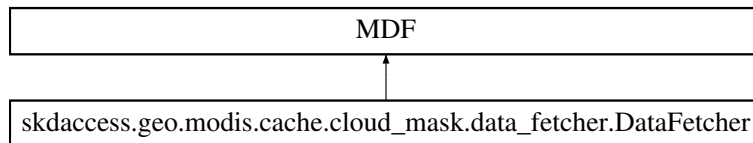
The documentation for this class was generated from the following file:

- [finance/timeseries/stream.py](#)

6.26 skdaccess.geo.modis.cache.cloud_mask.DataFetcher Class Reference

Data Fetcher for MODIS Cloud Mask.

Inheritance diagram for `skdaccess.geo.modis.cache.cloud_mask.DataFetcher`:



Public Member Functions

- `def __init__(self, ap_paramList, start_date, end_date, modis_platform='Terra', daynightboth='D', grid=None)`
Construct Data Fetcher for MODIS cloud mask data.

6.26.1 Detailed Description

Data Fetcher for MODIS Cloud Mask.

6.26.2 Constructor & Destructor Documentation

6.26.2.1 __init__()

```

def skdaccess.geo.modis.cache.cloud_mask.DataFetcher.__init__(
    self,
    ap_paramList,
    start_date,
    end_date,
    modis_platform = 'Terra',
    daynightboth = 'D',
    grid = None )
  
```

Construct Data Fetcher for MODIS cloud mask data.

Parameters

<i>ap_paramList[lat]</i>	Search latitude
<i>ap_paramList[lon]</i>	Search longitude
<i>start_date</i>	Starting date
<i>end_date</i>	Ending date
<i>modis_platform</i>	Platform (Either "Terra" or "Aqua")
<i>daynightboth</i>	Use daytime data ('D'), nighttime data ('N') or both ('B')
<i>grid</i>	Further divide each image into a multiple grids of size (y,x)

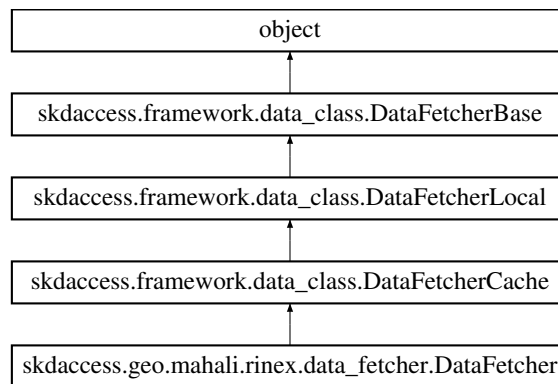
The documentation for this class was generated from the following file:

- [geo/modis/cache/cloud_mask/data_fetcher.py](#)

6.27 skdaccess.geo.mahali.rinex.DataFetcher Class Reference

Data Fetcher for Mahali Data.

Inheritance diagram for skdaccess.geo.mahali.rinex.DataFetcher:



Public Member Functions

- `def __init__(self, ap_paramList=[], start_date=None, end_date=None, generate_links=False)`
Initialize Mahali Data Fetcher.
- `def cacheData(self)`
Downloads all needed data.
- `def output(self)`
Generate data wrapper for Mahali data.
- `def checkIfDataExists(self, in_file_name)`
Checks if the file exists on the filesystem and the file is not empty.
- `def cacheData(self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)`

- *Download and store specified data to local disk.*
- def [multirun_enabled](#) (self)
Returns whether or not this data fetcher is multirun enabled.
- def [getHDFStorage](#) (self, keyname)
Retrieve a Pandas HDF Store for a dataset.
- def [getDataLocation](#) (data_name)
Get the location of data set.
- def [setDataLocation](#) (data_name, location, key='data_location')
Set the location of a data set.
- def [perturb](#) (self)
Perturb parameters.
- def [reset](#) (self)
Set all parameters to initial value.
- def [__str__](#) (self)
Generate string description.
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [start_date](#)
- [end_date](#)
- [date_range](#)
- [generate_links](#)
- [ap_paramList](#)
- [verbose](#)

6.27.1 Detailed Description

Data Fetcher for Mahali Data.

6.27.2 Constructor & Destructor Documentation

6.27.2.1 `__init__()`

```
def skdaccess.geo.mahali.rinex.DataFetcher.__init__ (
    self,
    ap_paramList = [],
    start_date = None,
    end_date = None,
    generate_links = False )
```

Initialize Mahali Data Fetcher.

Parameters

<i>ap_paramList[stations]</i>	Autolist of stations (Defaults to all stations)
<i>start_date</i>	Starting date for seelcting data (Defaults to beginning of available data)
<i>end_date</i>	Ending date for selecting data (Defaults to end of available data)
<i>generate_links</i>	Generate links to data instead of downloading data

6.27.3 Member Function Documentation

6.27.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.27.3.2 `cacheData()` [1/2]

```
def skdaccess.geo.mahali.rinex.DataFetcher.cacheData (
    self )
```

Downloads all needed data.

Called by [output\(\)](#).

6.27.3.3 cacheData() [2/2]

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.27.3.4 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.27.3.5 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.27.3.6 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.27.3.7 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.27.3.8 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.27.3.9 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.27.3.10 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.27.3.11 output()

```
def skdaccess.geo.mahali.rinex.DataFetcher.output (
    self )
```

Generate data wrapper for Mahali data.

Returns

Mahali data wrapper

6.27.3.12 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.27.3.13 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.27.3.14 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.27.3.15 `verbose_print()`

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.27.3.16 `writeConfig()`

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.27.3.17 `writeConfigItem()`

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.27.4 Member Data Documentation**6.27.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.27.4.2 date_range

`skdaccess.geo.mahali.rinex.DataFetcher.date_range`

6.27.4.3 end_date

`skdaccess.geo.mahali.rinex.DataFetcher.end_date`

6.27.4.4 generate_links

`skdaccess.geo.mahali.rinex.DataFetcher.generate_links`

6.27.4.5 start_date

`skdaccess.geo.mahali.rinex.DataFetcher.start_date`

6.27.4.6 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

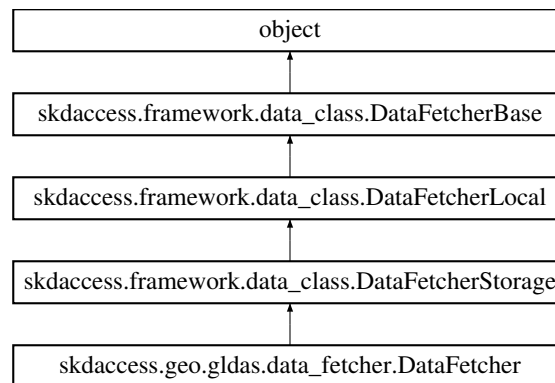
The documentation for this class was generated from the following file:

- [geo/mahali/rinex/data_fetcher.py](#)

6.28 skdaccess.geo.gldas.DataFetcher Class Reference

Data Fetcher for GLDAS data.

Inheritance diagram for `skdaccess.geo.gldas.DataFetcher`:



Public Member Functions

- `def __init__(self, ap_paramList, start_date=None, end_date=None, resample=False)`
Construct a GLDAS Data Fetcher.
- `def output(self)`
Create data wrapper of GLDAS data for specified geopoint.
- `def downloadFullDataset(cls, out_file=None, use_file=None)`
Download GLDAS data.
- `def __str__(self)`
String representation of data fetcher.
- `def multirun_enabled(self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getDataLocation(data_name)`
Get the location of data set.
- `def setDataLocation(data_name, location, key='data_location')`
Set the location of a data set.
- `def perturb(self)`
Perturb parameters.
- `def reset(self)`

- *Set all parameters to initial value.*
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [start_date](#)
- [end_date](#)
- [resample](#)
- [ap_paramList](#)
- [verbose](#)

6.28.1 Detailed Description

Data Fetcher for GLDAS data.

6.28.2 Constructor & Destructor Documentation

6.28.2.1 `__init__()`

```
def skdaccess.geo.gldas.DataFetcher.__init__ (
    self,
    ap_paramList,
    start_date = None,
    end_date = None,
    resample = False )
```

Construct a GLDAS Data Fetcher.

Parameters

<i>ap_paramList</i> [<i>geo_point</i>]	Autolist of Geographic location tuples
<i>start_date</i>	Beginning date
<i>end_date</i>	Ending date
<i>resample</i>	Resample the data to daily resolution, leaving NaN's in days without data (Default True)

6.28.3 Member Function Documentation

6.28.3.1 `__str__()`

```
def skdaccess.geo.gldas.DataFetcher.__str__ (
    self )
```

String representation of data fetcher.

Returns

String listing the name and geopoint of data fetcher

6.28.3.2 `downloadFullDataset()`

```
def skdaccess.geo.gldas.DataFetcher.downloadFullDataset (
    cls,
    out_file = None,
    use_file = None )
```

Download GLDAS data.

Parameters

<i>out_file</i>	Output filename for parsed data
<i>use_file</i>	Directory of downloaded data. If None, data will be downloaded.

Returns

Absolute path of parsed data

6.28.3.3 `getConfig()`

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

`configParser.ConfigParser` object of configuration

6.28.3.4 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.28.3.5 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.28.3.6 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.28.3.7 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStorage.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.28.3.8 output()

```
def skdaccess.geo.gldas.DataFetcher.output (
    self )
```

Create data wrapper of GLDAS data for specified geoint.

Returns

GLDAS Data Wrapper

6.28.3.9 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.28.3.10 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.28.3.11 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.28.3.12 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.28.3.13 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.28.3.14 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.28.4 Member Data Documentation**6.28.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.28.4.2 end_date

`skdaccess.geo.gldas.DataFetcher.end_date`

6.28.4.3 resample

`skdaccess.geo.gldas.DataFetcher.resample`

6.28.4.4 start_date

`skdaccess.geo.gldas.DataFetcher.start_date`

6.28.4.5 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

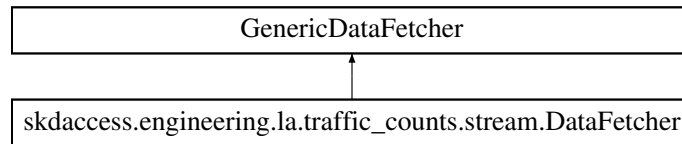
The documentation for this class was generated from the following file:

- [geo/gldas/data_fetcher.py](#)

6.29 skdaccess.engineering.la.traffic_counts.stream.DataFetcher Class Reference

[DataFetcher](#) for retrieving traffic counts from LA.

Inheritance diagram for skdaccess.engineering.la.traffic_counts.stream.DataFetcher:



Public Member Functions

- `def __init__(self, limit=None, start_time=None, end_time=None, app_token=None, verbose=False)`
Initialize Data Fetcher to retrieve traffic counts from LA.

6.29.1 Detailed Description

[DataFetcher](#) for retrieving traffic counts from LA.

6.29.2 Constructor & Destructor Documentation

6.29.2.1 __init__()

```

def skdaccess.engineering.la.traffic_counts.stream.DataFetcher.__init__(
    self,
    limit = None,
    start_time = None,
    end_time = None,
    app_token = None,
    verbose = False )

```

Initialize Data Fetcher to retrieve traffic counts from LA.

Parameters

<i>limit</i>	Maximum number of rows
<i>start_time</i>	Starting time
<i>end_time</i>	Ending time
<i>app_token</i>	Application token to avoid throttling
<i>verbose</i>	Print extra information

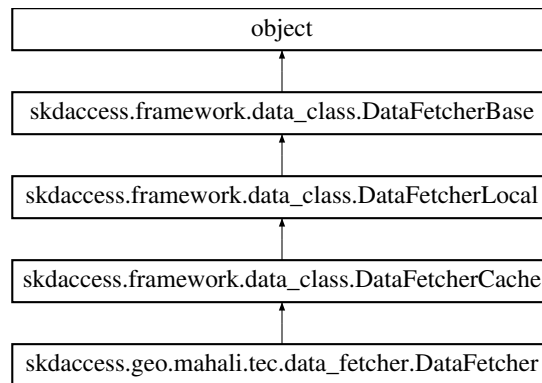
The documentation for this class was generated from the following file:

- [engineering/la/traffic_counts/stream.py](#)

6.30 skdaccess.geo.mahali.tec.DataFetcher Class Reference

Data Fetcher for Mahali Data.

Inheritance diagram for skdaccess.geo.mahali.tec.DataFetcher:



Public Member Functions

- `def __init__(self, ap_paramList=[], start_date=None, end_date=None)`
Initialize Mahali Data Fetcher.
- `def output(self)`
Generate data wrapper for Mahali tec data.
- `def checkIfDataExists(self, in_file_name)`
Checks if the file exists on the filesystem and the file is not empty.
- `def cacheData(self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)`
Download and store specified data to local disk.
- `def multirun_enabled(self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getHDFStorage(self, keyname)`
Retrieve a Pandas HDF Store for a dataset.
- `def getDataLocation(data_name)`
Get the location of data set.
- `def setDataLocation(data_name, location, key='data_location')`
Set the location of a data set.
- `def perturb(self)`
Perturb parameters.
- `def reset(self)`
Set all parameters to initial value.

- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- `start_date`
- `end_date`
- `date_range`
- `ap_paramList`
- `verbose`

6.30.1 Detailed Description

Data Fetcher for Mahali Data.

6.30.2 Constructor & Destructor Documentation

6.30.2.1 `__init__`()

```
def skdaccess.geo.mahali.tec.DataFetcher.__init__ (
    self,
    ap_paramList = [],
    start_date = None,
    end_date = None )
```

Initialize Mahali Data Fetcher.

Parameters

<code>ap_paramList[stations]</code>	Autolist of stations (Defaults to all stations)
<code>start_date</code>	Starting date for seelcting data (Defaults to beginning of available data)
<code>end_date</code>	Ending date for selecting data (Defaults to end of available data)

6.30.3 Member Function Documentation

6.30.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.30.3.2 `cacheData()`

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.30.3.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.30.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.30.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.30.3.6 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.30.3.7 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.30.3.8 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.30.3.9 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.30.3.10 output()

```
def skdaccess.geo.mahali.tec.DataFetcher.output (
    self )
```

Generate data wrapper for Mahali tec data.

Returns

Mahali data wrapper

6.30.3.11 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.30.3.12 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.30.3.13 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.30.3.14 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.30.3.15 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.30.3.16 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.30.4 Member Data Documentation**6.30.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.30.4.2 date_range

`skdaccess.geo.mahali.tec.DataFetcher.date_range`

6.30.4.3 end_date

`skdaccess.geo.mahali.tec.DataFetcher.end_date`

6.30.4.4 start_date

`skdaccess.geo.mahali.tec.DataFetcher.start_date`

6.30.4.5 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

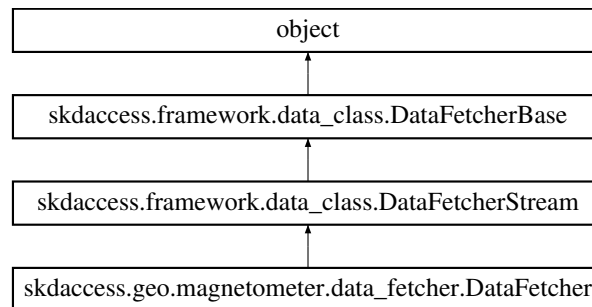
The documentation for this class was generated from the following file:

- `geo/mahali/tec/data_fetcher.py`

6.31 skdaccess.geo.magnetometer.DataFetcher Class Reference

Data fetcher for USGS geomagnetic observatories.

Inheritance diagram for skdaccess.geo.magnetometer.DataFetcher:



Public Member Functions

- def `__init__` (self, `ap_paramList`, `start_time`, `end_time`, `interval`='minute', `channels`=('X', 'Y', 'Z', 'F'), `data_type`='variation')
Geomagnetism Data fetcher constructor.
- def `output` (self)
Generate data wrapper for USGS geomagnetic data.
- def `getDataMetadata` ()
Get data metadata.
- def `retrieveOnlineData` (self, `data_specification`)
Method for downloading data into memory.
- def `multirun_enabled` (self)
Returns whether or not this data fetcher is multirun enabled.
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [start_time](#)
- [end_time](#)
- [interval](#)
- [channels](#)
- [data_type](#)
- [ap_paramList](#)
- [verbose](#)

6.31.1 Detailed Description

Data fetcher for USGS geomagnetic observatories.

6.31.2 Constructor & Destructor Documentation

6.31.2.1 `__init__()`

```
def skdaccess.geo.magnetometer.DataFetcher.__init__ (
    self,
    ap_paramList,
    start_time,
    end_time,
    interval = 'minute',
    channels = ('X', 'Y', 'Z', 'F'),
    data_type = 'variation' )
```

Geomagnetism Data fetcher constructor.

Parameters

<i>ap_paramList</i> [AutoList]	AutoList of Observatory names
<i>start_time</i>	Starting time
<i>end_time</i>	Ending time
<i>interval</i>	Time resolution
<i>channels</i>	Data channels
<i>data_type</i>	= Data type

6.31.3 Member Function Documentation

6.31.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.31.3.2 `getConfig()`

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.31.3.3 `getConfigItem()`

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.31.3.4 `getDataMetadata()`

```
def skdaccess.geo.magnetometer.DataFetcher.getDataMetadata ( )
```

Get data metadata.

Returns

Pandas dataframe containing station latitude and longitude coordinates

6.31.3.5 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.31.3.6 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.31.3.7 output()

```
def skdaccess.geo.magnetometer.DataFetcher.output (
    self )
```

Generate data wrapper for USGS geomagnetic data.

Returns

geomagnetic data wrapper

6.31.3.8 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.31.3.9 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.31.3.10 retrieveOnlineData()

```
def skdaccess.framework.data_class.DataFetcherStream.retrieveOnlineData (
    self,
    data_specification ) [inherited]
```

Method for downloading data into memory.

Parameters

<i>data_specification</i>	Url list of data to be retrieved
---------------------------	----------------------------------

Returns

Retrieved data

6.31.3.11 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.31.3.12 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.31.3.13 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.31.4 Member Data Documentation

6.31.4.1 `ap_paramList`

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.31.4.2 `channels`

`skdaccess.geo.magnetometer.DataFetcher.channels`

6.31.4.3 `data_type`

`skdaccess.geo.magnetometer.DataFetcher.data_type`

6.31.4.4 `end_time`

`skdaccess.geo.magnetometer.DataFetcher.end_time`

6.31.4.5 `interval`

`skdaccess.geo.magnetometer.DataFetcher.interval`

6.31.4.6 `start_time`

`skdaccess.geo.magnetometer.DataFetcher.start_time`

6.31.4.7 `verbose`

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

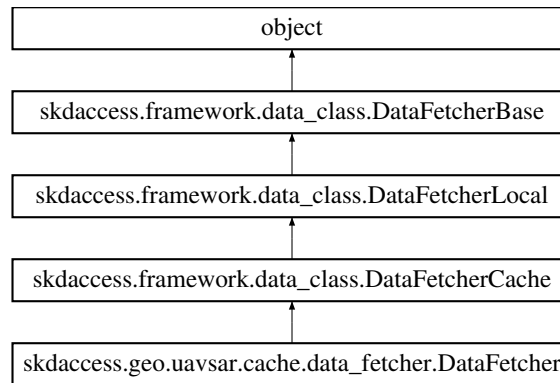
The documentation for this class was generated from the following file:

- `geo/magnetometer/data_fetcher.py`

6.32 skdaccess.geo.uavsar.cache.DataFetcher Class Reference

Data Fetcher for UAVSAR data.

Inheritance diagram for skdaccess.geo.uavsar.cache.DataFetcher:



Public Member Functions

- `def __init__ (self, slc_url_list, metadata_url_list, llh_url, memmap)`
Initialize UAVSAR data fetcher.
- `def output (self)`
Output data as a data wrapper.
- `def checkIfDataExists (self, in_file_name)`
Checks if the file exists on the filesystem and the file is not empty.
- `def cacheData (self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)`
Download and store specified data to local disk.
- `def multirun_enabled (self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getHDFStorage (self, keyname)`
Retrieve a Pandas HDF Store for a dataset.
- `def getDataLocation (data_name)`
Get the location of data set.
- `def setDataLocation (data_name, location, key='data_location')`
Set the location of a data set.
- `def perturb (self)`
Perturb parameters.
- `def reset (self)`
Set all parameters to initial value.
- `def __str__ (self)`
Generate string description.
- `def getMetadata (self)`
Return metadata about Data Fetcher.
- `def getConfig ()`

- *Retrieve skdaccess configuration.*
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [slc_url_list](#)
- [metadata_url_list](#)
- [llh_url](#)
- [memmap](#)
- [ap_paramList](#)
- [verbose](#)

6.32.1 Detailed Description

Data Fetcher for UAVSAR data.

6.32.2 Constructor & Destructor Documentation

6.32.2.1 `__init__()`

```
def skdaccess.geo.uavsar.cache.DataFetcher.__init__ (
    self,
    slc_url_list,
    metadata_url_list,
    llh_url,
    memmap )
```

Initialize UAVSAR data fetcher.

Parameters

<i>slc_url_list</i>	List of slc urls
<i>metadata_url_list</i>	List of metadata urls
<i>llh_url</i>	Latitude Longitude Height url
<i>memmap</i>	Open files using a memory map

6.32.3 Member Function Documentation

6.32.3.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.32.3.2 cacheData()

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.32.3.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.32.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.32.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.32.3.6 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.32.3.7 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.32.3.8 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.32.3.9 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.32.3.10 output()

```
def skdaccess.geo.uavsar.cache.DataFetcher.output (
    self )
```

Output data as a data wrapper.

Returns

Imagewrapper of data

6.32.3.11 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.32.3.12 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.32.3.13 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.32.3.14 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.32.3.15 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.32.3.16 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.32.4 Member Data Documentation**6.32.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.32.4.2 llh_url

`skdaccess.geo.uavsar.cache.DataFetcher.llh_url`

6.32.4.3 memmap

`skdaccess.geo.uavsar.cache.DataFetcher.memmap`

6.32.4.4 metadata_url_list

`skdaccess.geo.uavsar.cache.DataFetcher.metadata_url_list`

6.32.4.5 slc_url_list

`skdaccess.geo.uavsar.cache.DataFetcher.slc_url_list`

6.32.4.6 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

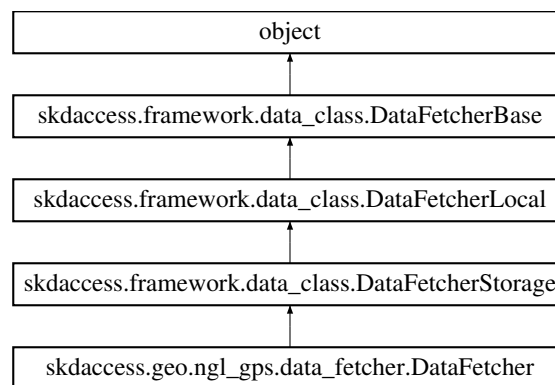
The documentation for this class was generated from the following file:

- `geo/uavsar/cache/data_fetcher.py`

6.33 skdaccess.geo.ngl_gps.DataFetcher Class Reference

Data fetcher for GPS data from Nevada Geodetic Laboratory.

Inheritance diagram for `skdaccess.geo.ngl_gps.DataFetcher`:



Public Member Functions

- `def __init__(self, start_date, end_date, lat_range, lon_range, mdyratio=0.7, data_type='ngl_gps')`
Construct NGL data fetcher.
- `def getStationMetadata()`
Get station metadata.
- `def getAntennaLogs()`
Retrieve information about antenna changes.
- `def output(self)`
Construct NGL GPS data wrapper.
- `def downloadFullDataset(cls, out_file, use_file=None)`
Abstract function used to download full data set.
- `def multirun_enabled(self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getDataLocation(data_name)`
Get the location of data set.
- `def setDataLocation(data_name, location, key='data_location')`
Set the location of a data set.
- `def perturb(self)`

- *Perturb parameters.*
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- `start_date`
- `end_date`
- `lat_range`
- `lon_range`
- `mdyratio`
- `data_type`
- `ap_paramList`
- `verbose`

6.33.1 Detailed Description

Data fetcher for GPS data from Nevada Geodetic Laboratory.

6.33.2 Constructor & Destructor Documentation

6.33.2.1 `__init__()`

```
def skdaccess.geo.ngl_gps.DataFetcher.__init__ (
    self,
    start_date,
    end_date,
    lat_range,
    lon_range,
    mdyratio = 0.7,
    data_type = 'ngl_gps' )
```

Construct NGL data fetcher.

Parameters

<i>start_date</i>	Starting date (string: '2002-01-01')
<i>end_date</i>	Ending date (string: '2015-01-01')
<i>lat_range</i>	Tuple containing latitude range
<i>lon_range</i>	Tuple containing longitude range
<i>mdyratio</i>	Choose stations whose ratio of valid/total is greater than mdyratio
<i>data_type</i>	Either 24 hour product ('ngl_gps') or 5 minute product ('ngl_5min')

6.33.3 Member Function Documentation

6.33.3.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.33.3.2 downloadFullDataset()

```
def skdaccess.framework.data_class.DataFetcherStorage.downloadFullDataset (
    cls,
    out_file,
    use_file = None ) [inherited]
```

Abstract function used to download full data set.

Parameters

<i>out_file</i>	output file name
<i>use_file</i>	Use previously downloaded data

Returns

Absolute path of parsed data

6.33.3.3 getAntennaLogs()

```
def skdaccess.geo.ngl_gps.DataFetcher.getAntennaLogs ( )
```

Retrieve information about antenna changes.

Returns

dictionary of antenna changes

6.33.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.33.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.33.3.6 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.33.3.7 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.33.3.8 getStationMetadata()

```
def skdaccess.geo.ngl_gps.DataFetcher.getStationMetadata ( )
```

Get station metadata.

Returns

data frame of station metadata

6.33.3.9 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStorage.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.33.3.10 output()

```
def skdaccess.geo.ngl_gps.DataFetcher.output (
    self )
```

Construct NGL GPS data wrapper.

Returns

NGL GPS data wrapper

6.33.3.11 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.33.3.12 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.33.3.13 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.33.3.14 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.33.3.15 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.33.3.16 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.33.4 Member Data Documentation**6.33.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.33.4.2 data_type

`skdaccess.geo.ngl_gps.DataFetcher.data_type`

6.33.4.3 end_date

`skdaccess.geo.ngl_gps.DataFetcher.end_date`

6.33.4.4 lat_range

`skdaccess.geo.ngl_gps.DataFetcher.lat_range`

6.33.4.5 lon_range

`skdaccess.geo.ngl_gps.DataFetcher.lon_range`

6.33.4.6 mdyratio

`skdaccess.geo.ngl_gps.DataFetcher.mdyratio`

6.33.4.7 start_date

`skdaccess.geo.ngl_gps.DataFetcher.start_date`

6.33.4.8 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

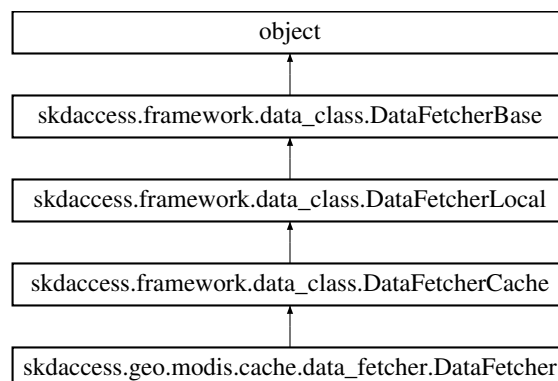
The documentation for this class was generated from the following file:

- [geo/ngl_gps/data_fetcher.py](#)

6.34 skdaccess.geo.modis.cache.DataFetcher Class Reference

Data Fetcher for MODIS data.

Inheritance diagram for `skdaccess.geo.modis.cache.DataFetcher`:



Public Member Functions

- def `__init__` (self, ap_paramList, modis_platform, modis_id, variable_list, start_date, end_date, daynightboth='D', grid=None, grid_fill=np.nan, use_long_name=False)
Construct Data Fetcher object.
- def `find_data` (self, fileid_list, file_object)
Finds files previously downloaded files associated with fileids.
- def `cacheData` (self, data_specification)
Download MODIS data.
- def `output` (self)
Generate data wrapper.
- def `checkIfDataExists` (self, in_file_name)
Checks if the file exists on the filesystem and the file is not empty.
- def `cacheData` (self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)
Download and store specified data to local disk.
- def `multirun_enabled` (self)
Returns whether or not this data fetcher is multirun enabled.
- def `getHDFStorage` (self, keyname)
Retrieve a Pandas HDF Store for a dataset.
- def `getDataLocation` (data_name)
Get the location of data set.
- def `setDataLocation` (data_name, location, key='data_location')
Set the location of a data set.
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [modis_id](#)
- [variable_list](#)
- [start_date](#)
- [end_date](#)
- [daynightboth](#)
- [grid](#)
- [grid_fill](#)
- [use_long_name](#)
- [modis_platform](#)
- [modis_identifier](#)
- [ap_paramList](#)
- [verbose](#)

6.34.1 Detailed Description

Data Fetcher for MODIS data.

6.34.2 Constructor & Destructor Documentation

6.34.2.1 `__init__()`

```
def skdaccess.geo.modis.cache.DataFetcher.__init__ (
    self,
    ap_paramList,
    modis_platform,
    modis_id,
    variable_list,
    start_date,
    end_date,
    daynightboth = 'D',
    grid = None,
    grid_fill = np.nan,
    use_long_name = False )
```

Construct Data Fetcher object.

Parameters

<i>ap_paramList[lat]</i>	Search latitude
<i>ap_paramList[lon]</i>	Search longitude
<i>modis_platform</i>	Platform (Either "Terra" or "Aqua")
<i>modis_id</i>	Product string (e.g. '06_L2')
<i>variable_list</i>	List of variables to fetch

Parameters

<i>start_date</i>	Starting date
<i>end_date</i>	Ending date
<i>daynightboth</i>	Use daytime data ('D'), nighttime data ('N') or both ('B')
<i>grid</i>	Further divide each image into a multiple grids of size (y,x)
<i>grid_fill</i>	Fill value to use when creating gridded data
<i>use_long_name</i>	Use long names for metadata instead of variable name

6.34.3 Member Function Documentation

6.34.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.34.3.2 `cacheData()` [1/2]

```
def skdaccess.geo.modis.cache.DataFetcher.cacheData (
    self,
    data_specification )
```

Download MODIS data.

Parameters

<i>data_specification</i>	List of file IDs to cache
---------------------------	---------------------------

6.34.3.3 `cacheData()` [2/2]

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
```

```

username = None,
password = None,
authentication_url = None,
cookiejar = None,
use_requests = False,
use_progress_bar = True ) [inherited]

```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.34.3.4 checkIfDataExists()

```

def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]

```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.34.3.5 find_data()

```
def skdaccess.geo.modis.cache.DataFetcher.find_data (
    self,
    fileid_list,
    file_object )
```

Finds files previously downloaded files associated with fileids.

Parameters

<i>fileid_list</i>	List of file id's
<i>file_object</i>	File object to read from

Returns

Pandas series of file locaitons indexed by file id

6.34.3.6 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.34.3.7 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.34.3.8 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.34.3.9 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.34.3.10 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.34.3.11 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.34.3.12 output()

```
def skdaccess.geo.modis.cache.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

data wrapper of MODIS data

6.34.3.13 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.34.3.14 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.34.3.15 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.34.3.16 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.34.3.17 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.34.3.18 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.34.4 Member Data Documentation**6.34.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.34.4.2 daynightboth

`skdaccess.geo.modis.cache.DataFetcher.daynightboth`

6.34.4.3 end_date

`skdaccess.geo.modis.cache.DataFetcher.end_date`

6.34.4.4 grid

`skdaccess.geo.modis.cache.DataFetcher.grid`

6.34.4.5 grid_fill

`skdaccess.geo.modis.cache.DataFetcher.grid_fill`

6.34.4.6 modis_id

`skdaccess.geo.modis.cache.DataFetcher.modis_id`

6.34.4.7 modis_identifier

`skdaccess.geo.modis.cache.DataFetcher.modis_identifier`

6.34.4.8 modis_platform

`skdaccess.geo.modis.cache.DataFetcher.modis_platform`

6.34.4.9 start_date

`skdaccess.geo.modis.cache.DataFetcher.start_date`

6.34.4.10 use_long_name

`skdaccess.geo.modis.cache.DataFetcher.use_long_name`

6.34.4.11 variable_list

`skdaccess.geo.modis.cache.DataFetcher.variable_list`

6.34.4.12 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

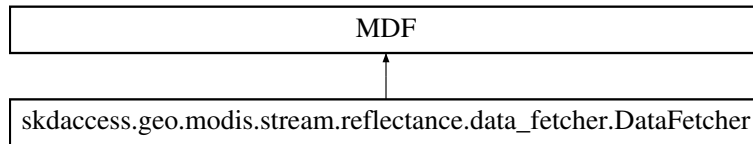
The documentation for this class was generated from the following file:

- `geo/modis/cache/data_fetcher.py`

6.35 skdaccess.geo.modis.stream.reflectance.DataFetcher Class Reference

Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

Inheritance diagram for skdaccess.geo.modis.stream.reflectance.DataFetcher:



Public Member Functions

- `def __init__ (self, ap_paramList, start_date, end_date, modis_platform='Terra', daynightboth='D', grid=None, bands=[1])`

Construct Data Fetcher for MODIS 1km surface reflectance.

6.35.1 Detailed Description

Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

6.35.2 Constructor & Destructor Documentation

6.35.2.1 __init__()

```

def skdaccess.geo.modis.stream.reflectance.DataFetcher.__init__ (
    self,
    ap_paramList,
    start_date,
    end_date,
    modis_platform = 'Terra',
    daynightboth = 'D',
    grid = None,
    bands = [1 ]
)
  
```

Construct Data Fetcher for MODIS 1km surface reflectance.

Parameters

<code>ap_paramList[lat]</code>	Search latitude
<code>ap_paramList[lon]</code>	Search longitude
<code>start_date</code>	Starting date
<code>end_date</code>	Ending date
<code>modis_platform</code>	Platform (Either "Terra" or "Aqua")
<code>daynightboth</code>	Use daytime data ('D'), nighttime data ('N') or both ('B')
<code>grid</code>	Further divide each image into a multiple grids of size (y,x)

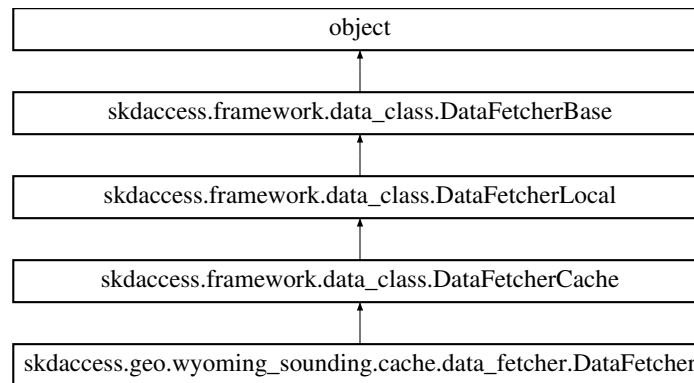
The documentation for this class was generated from the following file:

- [geo/modis/stream/reflectance/data_fetcher.py](#)

6.36 skdaccess.geo.wyoming_sounding.cache.DataFetcher Class Reference

[DataFetcher](#) for retrieving Wyoming Sounding data.

Inheritance diagram for skdaccess.geo.wyoming_sounding.cache.DataFetcher:



Public Member Functions

- `def __init__(self, station_number, year, month, day_start, day_end, start_hour=0, end_hour=12)`
Initialize Data Fetcher.
- `def output(self)`
Generate data wrapper.
- `def checkIfDataExists(self, in_file_name)`
Checks if the file exists on the filesystem and the file is not empty.
- `def cacheData(self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)`
Download and store specified data to local disk.
- `def multirun_enabled(self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getHDFStorage(self, keyname)`
Retrieve a Pandas HDF Store for a dataset.
- `def getDataLocation(data_name)`
Get the location of data set.
- `def setDataLocation(data_name, location, key='data_location')`
Set the location of a data set.
- `def perturb(self)`
Perturb parameters.
- `def reset(self)`
Set all parameters to initial value.

- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- `station_number`
- `year_list`
- `month_list`
- `day_start`
- `day_end`
- `start_hour`
- `end_hour`
- `ap_paramList`
- `verbose`

6.36.1 Detailed Description

`DataFetcher` for retrieving Wyoming Sounding data.

6.36.2 Constructor & Destructor Documentation

6.36.2.1 `__init__()`

```
def skdaccess.geo.wyoming_sounding.cache.DataFetcher.__init__ (
    self,
    station_number,
    year,
    month,
    day_start,
    day_end,
    start_hour = 0,
    end_hour = 12 )
```

Initialize Data Fetcher.

Parameters

<i>station_number</i>	Station number
<i>year</i>	Input year
<i>month</i>	Input month (Integer for a single month, or a list of integers for multiple months)
<i>day_start</i>	First day of the month to include
<i>day_end</i>	Last day of the month to include
<i>start_hour</i>	Starting hour (may be either 0 or 12)
<i>end_hour</i>	Ending hour (may be either 0 or 12)

6.36.3 Member Function Documentation

6.36.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.36.3.2 `cacheData()`

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.36.3.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.36.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.36.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.36.3.6 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.36.3.7 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.36.3.8 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.36.3.9 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.36.3.10 output()

```
def skdaccess.geo.wyoming_sounding.cache.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

Wyoming sounding data in a data wrapper

6.36.3.11 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.36.3.12 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.36.3.13 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.36.3.14 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.36.3.15 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.36.3.16 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.36.4 Member Data Documentation

6.36.4.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]
```

6.36.4.2 day_end

```
skdaccess.geo.wyoming_sounding.cache.DataFetcher.day_end
```

6.36.4.3 day_start

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.day_start`

6.36.4.4 end_hour

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.end_hour`

6.36.4.5 month_list

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.month_list`

6.36.4.6 start_hour

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.start_hour`

6.36.4.7 station_number

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.station_number`

6.36.4.8 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

6.36.4.9 year_list

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.year_list`

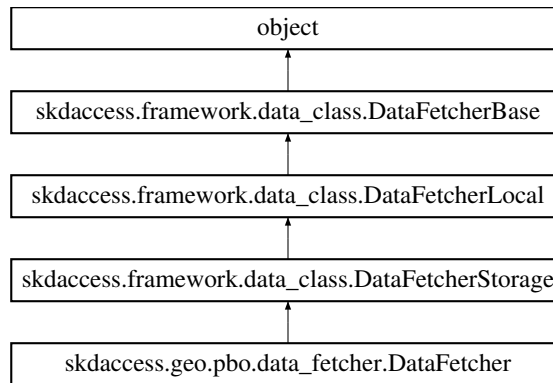
The documentation for this class was generated from the following file:

- `geo/wyoming_sounding/cache/data_fetcher.py`

6.37 skdaccess.geo.pbo.DataFetcher Class Reference

Data fetcher for PBO GPS data.

Inheritance diagram for skdaccess.geo.pbo.DataFetcher:



Public Member Functions

- def `__init__` (self, start_time, end_time, [ap_paramList](#), mdyratio=.5, [default_columns](#)=['dN', dE, dU, [default_↵](#)
[error_columns](#)=['Sn', Se, Su, [use_progress_bar](#)=True, [index_date_only](#)=True)
Initialize a [DataFetcher](#).
- def [setStationList](#) (self, [station_list](#))
Set the list of stations to use.
- def [getInfo](#) (self)
Get information about the stations and geo_point.
- def [output](#) (self)
Generate PBO Data Wrapper.
- def `__str__` (self)
print the parameter values
- def [getStationMetadata](#) (data_frame=False)
Read in the metadata and convert to dictionary.
- def [getAntennaLogs](#) ()
Get antenna logs.
- def [downloadFullDataset](#) (cls, out_file='pbo_data.h5', use_file=None)
Download and parse data from the Plate Boundary Observatory.
- def [multirun_enabled](#) (self)
Returns whether or not this data fetcher is multirun enabled.
- def [getDataLocation](#) (data_name)
Get the location of data set.
- def [setDataLocation](#) (data_name, location, key='data_location')
Set the location of a data set.
- def [perturb](#) (self)
Perturb parameters.
- def [reset](#) (self)

Set all parameters to initial value.

- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [station_list](#)
- [default_columns](#)
- [default_error_columns](#)
- [use_progress_bar](#)
- [index_date_only](#)
- [antenna_info](#)
- [meta_data](#)
- [ap_paramList](#)
- [verbose](#)

6.37.1 Detailed Description

Data fetcher for PBO GPS data.

6.37.2 Constructor & Destructor Documentation

6.37.2.1 `__init__()`

```
def skdaccess.geo.pbo.DataFetcher.__init__ (
    self,
    start_time,
    end_time,
    ap_paramList,
    mdyratio = .5,
    default_columns = ['dN',
    dE,
    dU,
    default_error_columns = ['Sn',
    Se,
    Su,
    use_progress_bar = True,
    index_date_only = True )
```

Initialize a [DataFetcher](#).

Parameters

<i>start_time</i>	String of starting date in the form of "2005-01-01"
<i>end_time</i>	String of ending date in the form of "2014-12-31"
<i>ap_paramList[lat_range]</i>	AutoList, Latitude range used to select stabilization sites
<i>ap_paramList[lon_range]</i>	AutoList, Longitude range used to select stabilization sites
<i>mdyratio</i>	Only keep stations that have mdyratio of data in the specified time range
<i>default_columns</i>	Default columns to process
<i>default_error_columns</i>	Default error columns to process
<i>use_progress_bar</i>	Use a progress bar when loading data
<i>index_date_only</i>	Create a index using date only (no hour information)

6.37.3 Member Function Documentation

6.37.3.1 __str__()

```
def skdaccess.geo.pbo.DataFetcher.__str__ (
    self )
```

print the parameter values

Returns

String representation of Data Fetcher

6.37.3.2 downloadFullDataset()

```
def skdaccess.geo.pbo.DataFetcher.downloadFullDataset (
    cls,
    out_file = 'pbo_data.h5',
    use_file = None )
```

Download and parse data from the Plate Boundary Observatory.

Parameters

<i>out_file</i>	Output filename for parsed data
<i>use_file</i>	Use already downloaded data. If None, data will be downloaded.

Returns

Absolute path of parsed data

6.37.3.3 getAntennaLogs()

```
def skdaccess.geo.pbo.DataFetcher.getAntennaLogs ( )
```

Get antenna logs.

Returns

dictionary of data frames containing antenna logs

6.37.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.37.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.37.3.6 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.37.3.7 getInfo()

```
def skdaccess.geo.pbo.DataFetcher.getInfo (
    self )
```

Get information about the stations and geo_point.

Returns

tuple containing station list and geo_point

6.37.3.8 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.37.3.9 getStationMetadata()

```
def skdaccess.geo.pbo.DataFetcher.getStationMetadata (
    data_frame = False )
```

Read in the metadata and convert to dictionary.

Returns

dictionary of PBO metadata

6.37.3.10 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStorage.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.37.3.11 output()

```
def skdaccess.geo.pbo.DataFetcher.output (
    self )
```

Generate PBO Data Wrapper.

Returns

PBO Data Wrapper

6.37.3.12 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.37.3.13 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.37.3.14 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.37.3.15 setStationList()

```
def skdaccess.geo.pbo.DataFetcher.setStationList (
    self,
    station_list )
```

Set the list of stations to use.

Parameters

<i>station_list</i>	List of stations to fetch
---------------------	---------------------------

6.37.3.16 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
```

```
    args,  
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.37.3.17 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (  
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.37.3.18 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (  
    section,  
    key,  
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.37.4 Member Data Documentation

6.37.4.1 antenna_info

`skdaccess.geo.pbo.DataFetcher.antenna_info`

6.37.4.2 ap_paramList

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.37.4.3 default_columns

`skdaccess.geo.pbo.DataFetcher.default_columns`

6.37.4.4 default_error_columns

`skdaccess.geo.pbo.DataFetcher.default_error_columns`

6.37.4.5 index_date_only

`skdaccess.geo.pbo.DataFetcher.index_date_only`

6.37.4.6 meta_data

`skdaccess.geo.pbo.DataFetcher.meta_data`

6.37.4.7 station_list

`skdaccess.geo.pbo.DataFetcher.station_list`

6.37.4.8 use_progress_bar

`skdaccess.geo.pbo.DataFetcher.use_progress_bar`

6.37.4.9 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

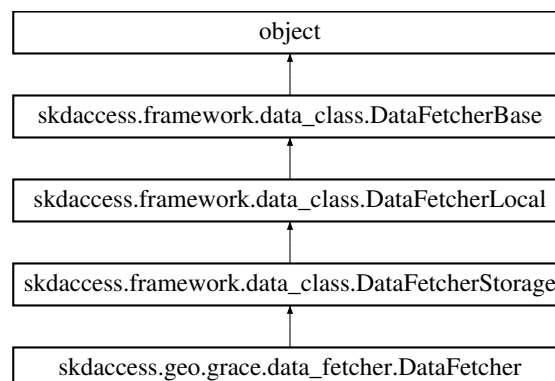
The documentation for this class was generated from the following file:

- [geo/pbo/data_fetcher.py](#)

6.38 skdaccess.geo.grace.DataFetcher Class Reference

Data Fetcher for GRACE data.

Inheritance diagram for `skdaccess.geo.grace.DataFetcher`:



Public Member Functions

- def [__init__](#) (self, [ap_paramList](#), [start_date](#)=None, [end_date](#)=None)
Construct a Grace Data Fetcher.
- def [output](#) (self)
Create data wrapper of grace data for specified geopoints.
- def [__str__](#) (self)
String representation of data fetcher.
- def [downloadFullDataset](#) (cls, out_file='grace.h5', use_file=None)
Download and parse data from the Gravity Recovery and Climate Experiment.
- def [multirun_enabled](#) (self)
Returns whether or not this data fetcher is multirun enabled.
- def [getDataLocation](#) (data_name)
Get the location of data set.
- def [setDataLocation](#) (data_name, location, key='data_location')
Set the location of a data set.
- def [perturb](#) (self)
Perturb parameters.
- def [reset](#) (self)
Set all parameters to initial value.
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [start_date](#)
- [end_date](#)
- [ap_paramList](#)
- [verbose](#)

6.38.1 Detailed Description

Data Fetcher for GRACE data.

6.38.2 Constructor & Destructor Documentation

6.38.2.1 `__init__()`

```
def skdaccess.geo.grace.DataFetcher.__init__ (
    self,
    ap_paramList,
    start_date = None,
    end_date = None )
```

Construct a Grace Data Fetcher.

Parameters

<i>ap_paramList</i> [<i>geo_point</i>]	AutoList of geographic location tuples (lat,lon)
<i>start_date</i>	Beginning date
<i>end_date</i>	Ending date

6.38.3 Member Function Documentation

6.38.3.1 `__str__()`

```
def skdaccess.geo.grace.DataFetcher.__str__ (
    self )
```

String representation of data fetcher.

Returns

String listing the name and geopoint of data fetcher

6.38.3.2 `downloadFullDataset()`

```
def skdaccess.geo.grace.DataFetcher.downloadFullDataset (
    cls,
    out_file = 'grace.h5',
    use_file = None )
```

Download and parse data from the Gravity Recovery and Climate Experiment.

Parameters

<i>out_file</i>	Output filename for parsed data
<i>use_file</i>	Directory of already downloaded data. If None, data will be downloaded.

Returns

Absolute path of parsed data

6.38.3.3 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.38.3.4 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.38.3.5 `getDataLocation()`

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.38.3.6 `getMetadata()`

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.38.3.7 `multirun_enabled()`

```
def skdaccess.framework.data_class.DataFetcherStorage.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.38.3.8 output()

```
def skdaccess.geo.grace.DataFetcher.output (
    self )
```

Create data wrapper of grace data for specified geopoints.

Returns

Grace Data Wrapper

6.38.3.9 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.38.3.10 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.38.3.11 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.38.3.12 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.38.3.13 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.38.3.14 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.38.4 Member Data Documentation

6.38.4.1 ap_paramList

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.38.4.2 end_date

`skdaccess.geo.grace.DataFetcher.end_date`

6.38.4.3 start_date

`skdaccess.geo.grace.DataFetcher.start_date`

6.38.4.4 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

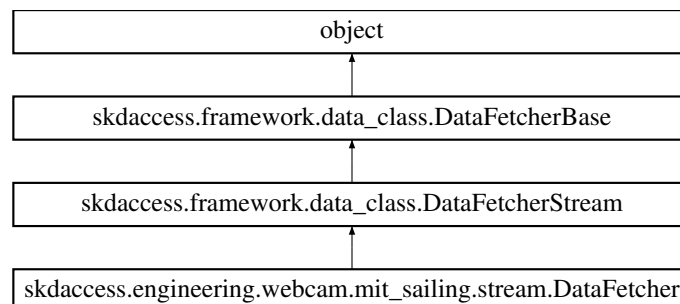
The documentation for this class was generated from the following file:

- [geo/grace/data_fetcher.py](#)

6.39 skdaccess.engineering.webcam.mit_sailing.stream.DataFetcher Class Reference

Data Fetcher for retrieving webcam images from the MIT Sailing Pavilion.

Inheritance diagram for `skdaccess.engineering.webcam.mit_sailing.stream.DataFetcher`:



Public Member Functions

- def `__init__` (self, `camera_list`=['E', SE, SW, W])
- def `output` (self)
Retrieve data from webcams at the MIT Sailing Pavilion.
- def `retrieveOnlineData` (self, `data_specification`)
Method for downloading data into memory.
- def `multirun_enabled` (self)
Returns whether or not this data fetcher is multirun enabled.
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- `camera_list`
- `ap_paramList`
- `verbose`

6.39.1 Detailed Description

Data Fetcher for retrieving webcam images from the MIT Sailing Pavilion.

6.39.2 Constructor & Destructor Documentation

6.39.2.1 `__init__()`

```
def skdaccess.engineering.webcam.mit_sailing.stream.DataFetcher.__init__ (
    self,
    camera_list = ['E',
                  SE,
                  SW,
                  W ]
)
```

Parameters

<i>camera_list</i>	Which camera to retrieve from (List that contains one or more of the following: 'E', 'SE', 'SW', or 'W')
--------------------	--

6.39.3 Member Function Documentation

6.39.3.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.39.3.2 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.39.3.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.39.3.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.39.3.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.39.3.6 output()

```
def skdaccess.engineering.webcam.mit_sailing.stream.DataFetcher.output (
    self )
```

Retrieve data from webcams at the MIT Sailing Pavilion.

Returns

Image Wrapper containing the latest images from the webcams

6.39.3.7 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.39.3.8 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.39.3.9 retrieveOnlineData()

```
def skdaccess.framework.data_class.DataFetcherStream.retrieveOnlineData (
    self,
    data_specification ) [inherited]
```

Method for downloading data into memory.

Parameters

<i>data_specification</i>	Url list of data to be retrieved
---------------------------	----------------------------------

Returns

Retrieved data

6.39.3.10 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.39.3.11 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.39.3.12 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.39.4 Member Data Documentation

6.39.4.1 ap_paramList

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.39.4.2 camera_list

`skdaccess.engineering.webcam.mit_sailing.stream.DataFetcher.camera_list`

6.39.4.3 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

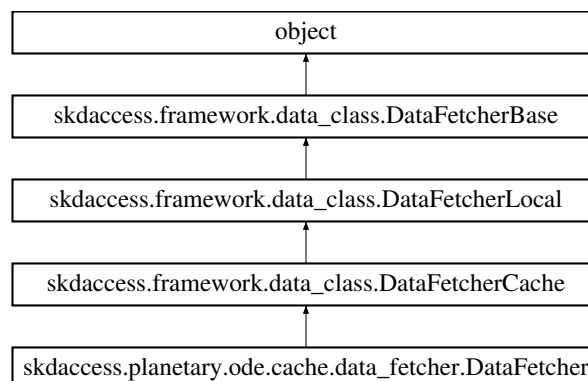
The documentation for this class was generated from the following file:

- `engineering/webcam/mit_sailing/stream.py`

6.40 skdaccess.planetary.ode.cache.DataFetcher Class Reference

Data Fetcher from the Orbital Data Explorer (ODE)

Inheritance diagram for `skdaccess.planetary.ode.cache.DataFetcher`:



Public Member Functions

- def `__init__` (self, `target`, `mission`, `instrument`, `product_type`, `western_lon`=None, `eastern_lon`=None, `min_lat`=None, `max_lat`=None, `min_ob_time`=", `max_ob_time`", `product_id`", `file_name`=' *', `number_product_limit`=10, `result_offset_number`=0, `remove_ndv`=True)
- def `output` (self)
Generate data wrapper from ODE data.
- def `checkIfDataExists` (self, `in_file_name`)
Checks if the file exists on the filesystem and the file is not empty.
- def `cacheData` (self, `keyname`, `online_path_list`, `username`=None, `password`=None, `authentication_url`=None, `cookiejar`=None, `use_requests`=False, `use_progress_bar`=True)
Download and store specified data to local disk.
- def `multirun_enabled` (self)
Returns whether or not this data fetcher is multirun enabled.
- def `getHDFStorage` (self, `keyname`)
Retrieve a Pandas HDF Store for a dataset.
- def `getDataLocation` (`data_name`)
Get the location of data set.
- def `setDataLocation` (`data_name`, `location`, `key`='data_location')
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (`section`, `key`)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (`section`, `key`, `value`)
Retrieve skdaccess configuration item.
- def `writeConfig` (`conf`)
Write config to disk.
- def `verbose_print` (self, `args`, `kwargs`)
Print statement if verbose flag is set.

Public Attributes

- `target`
- `mission`
- `instrument`
- `product_type`
- `western_lon`
- `eastern_lon`

- [min_lat](#)
- [max_lat](#)
- [min_ob_time](#)
- [max_ob_time](#)
- [product_id](#)
- [file_name](#)
- [number_product_limit](#)
- [result_offset_number](#)
- [remove_ndv](#)
- [ap_paramList](#)
- [verbose](#)

6.40.1 Detailed Description

Data Fetcher from the Orbital Data Explorer (ODE)

6.40.2 Constructor & Destructor Documentation

6.40.2.1 `__init__()`

```
def skdaccess.planetary.ode.cache.DataFetcher.__init__ (
    self,
    target,
    mission,
    instrument,
    product_type,
    western_lon = None,
    eastern_lon = None,
    min_lat = None,
    max_lat = None,
    min_ob_time = '',
    max_ob_time = '',
    product_id = '',
    file_name = '*',
    number_product_limit = 10,
    result_offset_number = 0,
    remove_ndv = True )
```

6.40.3 Member Function Documentation

6.40.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.40.3.2 `cacheData()`

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.40.3.3 `checkIfDataExists()`

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.40.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.40.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.40.3.6 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.40.3.7 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.40.3.8 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.40.3.9 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.40.3.10 output()

```
def skdaccess.planetary.ode.cache.DataFetcher.output (
    self )
```

Generate data wrapper from ODE data.

6.40.3.11 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.40.3.12 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.40.3.13 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.40.3.14 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.40.3.15 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.40.3.16 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.40.4 Member Data Documentation**6.40.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.40.4.2 eastern_lon

`skdaccess.planetary.ode.cache.DataFetcher.eastern_lon`

6.40.4.3 file_name

`skdaccess.planetary.ode.cache.DataFetcher.file_name`

6.40.4.4 instrument

`skdaccess.planetary.ode.cache.DataFetcher.instrument`

6.40.4.5 max_lat

`skdaccess.planetary.ode.cache.DataFetcher.max_lat`

6.40.4.6 max_ob_time

skdaccess.planetary.ode.cache.DataFetcher.max_ob_time

6.40.4.7 min_lat

skdaccess.planetary.ode.cache.DataFetcher.min_lat

6.40.4.8 min_ob_time

skdaccess.planetary.ode.cache.DataFetcher.min_ob_time

6.40.4.9 mission

skdaccess.planetary.ode.cache.DataFetcher.mission

6.40.4.10 number_product_limit

skdaccess.planetary.ode.cache.DataFetcher.number_product_limit

6.40.4.11 product_id

skdaccess.planetary.ode.cache.DataFetcher.product_id

6.40.4.12 product_type

skdaccess.planetary.ode.cache.DataFetcher.product_type

6.40.4.13 remove_ndv

```
skdaccess.planetary.ode.cache.DataFetcher.remove_ndv
```

6.40.4.14 result_offset_number

```
skdaccess.planetary.ode.cache.DataFetcher.result_offset_number
```

6.40.4.15 target

```
skdaccess.planetary.ode.cache.DataFetcher.target
```

6.40.4.16 verbose

```
skdaccess.framework.data_class.DataFetcherBase.verbose [inherited]
```

6.40.4.17 western_lon

```
skdaccess.planetary.ode.cache.DataFetcher.western_lon
```

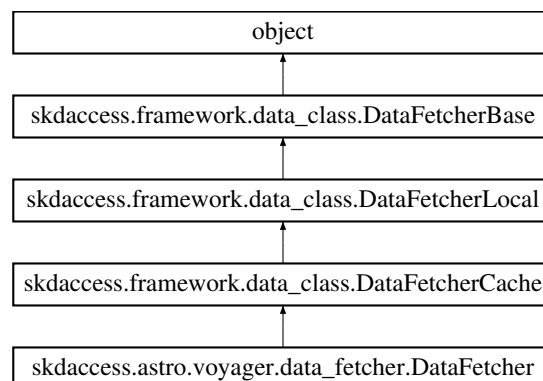
The documentation for this class was generated from the following file:

- [planetary/ode/cache/data_fetcher.py](#)

6.41 skdaccess.astro.voyager.DataFetcher Class Reference

Data Fetcher for Mahali temperature data.

Inheritance diagram for skdaccess.astro.voyager.DataFetcher:



Public Member Functions

- def `__init__` (self, start_year, end_year, spacecraft='both')
Initialize Voyager data fetcher.
- def `generateURL` (self, spacecraft, in_year)
Generate url for voyager data.
- def `parseVoyagerData` (self, spacecraft, in_filename)
Parse Voyager Data.
- def `parseVoyagerMetadata` (self, in_file)
Parse voyager metadata.
- def `getMetadataFiles` (self)
Get path to metadata file.
- def `output` (self)
Generate data wrapper.
- def `checkIfDataExists` (self, in_file_name)
Checks if the file exists on the filesystem and the file is not empty.
- def `cacheData` (self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)
Download and store specified data to local disk.
- def `multirun_enabled` (self)
Returns whether or not this data fetcher is multirun enabled.
- def `getHDFSStorage` (self, keyname)
Retrieve a Pandas HDF Store for a dataset.
- def `getDataLocation` (data_name)
Get the location of data set.
- def `setDataLocation` (data_name, location, key='data_location')
Set the location of a data set.
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [year_list](#)
- [spacecraft_list](#)
- [field_names](#)
- [field_widths](#)
- [base_url](#)
- [ap_paramList](#)
- [verbose](#)

6.41.1 Detailed Description

Data Fetcher for Mahali temperature data.

6.41.2 Constructor & Destructor Documentation

6.41.2.1 `__init__()`

```
def skdaccess.astro.voyager.DataFetcher.__init__ (
    self,
    start_year,
    end_year,
    spacecraft = 'both' )
```

Initialize Voyager data fetcher.

Parameters

<i>start_year</i>	Starting year
<i>end_year</i>	Ending year
<i>spacecraft</i>	Which spacecraft to use (voyager1, voyager2, or both).

6.41.3 Member Function Documentation

6.41.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.41.3.2 cacheData()

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.41.3.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.41.3.4 generateURL()

```
def skdaccess.astro.voyager.DataFetcher.generateURL (
    self,
    spacecraft,
    in_year )
```

Generate url for voyager data.

Parameters

<i>spacecraft</i>	Voyager spacecraft (vy1 or vy2)
<i>in_year</i>	Input year (or 'metadata')

Returns

Url of data location

6.41.3.5 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.41.3.6 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.41.3.7 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.41.3.8 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.41.3.9 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.41.3.10 getMetadataFiles()

```
def skdaccess.astro.voyager.DataFetcher.getMetadataFiles (
    self )
```

Get path to metadata file.

Metadata will download if necessary

Returns

List containing file path(s) for the metadata

6.41.3.11 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.41.3.12 output()

```
def skdaccess.astro.voyager.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

data wrapper of voyager data

6.41.3.13 parseVoyagerData()

```
def skdaccess.astro.voyager.DataFetcher.parseVoyagerData (
    self,
    spacecraft,
    in_filename )
```

Parse Voyager Data.

Parameters

<i>spacecraft</i>	Voyager spacecraft (vy1 or vy2)
<i>in_filename</i>	Input voyager data filename

Returns

Pandas Dataframe of Voyager data

6.41.3.14 parseVoyagerMetadata()

```
def skdaccess.astro.voyager.DataFetcher.parseVoyagerMetadata (
    self,
    in_file )
```

Parse voyager metadata.

Parameters

<i>in_file</i>	Input filename
----------------	----------------

Returns

Dictionary containing metadata

6.41.3.15 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.41.3.16 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.41.3.17 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.41.3.18 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
```

```
    args,  
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.41.3.19 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (  
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.41.3.20 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (  
    section,  
    key,  
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.41.4 Member Data Documentation

6.41.4.1 `ap_paramList`

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.41.4.2 `base_url`

`skdaccess.astro.voyager.DataFetcher.base_url`

6.41.4.3 `field_names`

`skdaccess.astro.voyager.DataFetcher.field_names`

6.41.4.4 `field_widths`

`skdaccess.astro.voyager.DataFetcher.field_widths`

6.41.4.5 `spacecraft_list`

`skdaccess.astro.voyager.DataFetcher.spacecraft_list`

6.41.4.6 `verbose`

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

6.41.4.7 year_list

```
skdaccess.astro.voyager.DataFetcher.year_list
```

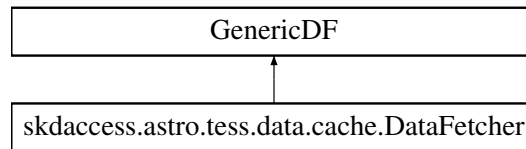
The documentation for this class was generated from the following file:

- astro/voyager/[data_fetcher.py](#)

6.42 skdaccess.astro.tess.data.cache.DataFetcher Class Reference

Data Fetcher for TESS data alerts.

Inheritance diagram for skdaccess.astro.tess.data.cache.DataFetcher:



Public Member Functions

- def `__init__` (self, [ap_paramList](#))
Initialize TESS Data Fetcher.
- def [generateURLFromTID](#) (self, tid_list)
Generate URL from TID.
- def [getTargetInformation](#) ()
Retrieve Target information for TESS Data Alerts.

Public Attributes

- [start_url](#)
- [end_url](#)

6.42.1 Detailed Description

Data Fetcher for TESS data alerts.

6.42.2 Constructor & Destructor Documentation

6.42.2.1 `__init__`()

```
def skdaccess.astro.tess.data.cache.DataFetcher.__init__ (
    self,
    ap_paramList )
```

Initialize TESS Data Fetcher.

Parameters

<code>ap_paramList[tess_ids]</code>	List of TESS IDs to retrieve
-------------------------------------	------------------------------

6.42.3 Member Function Documentation**6.42.3.1 generateURLFromTID()**

```
def skdaccess.astro.tess.data.cache.DataFetcher.generateURLFromTID (
    self,
    tid_list )
```

Generate URL from TID.

Parameters

<code>tid_list</code>	List of input Tess IDs
-----------------------	------------------------

6.42.3.2 getTargetInformation()

```
def skdaccess.astro.tess.data.cache.DataFetcher.getTargetInformation ( )
```

Retrieve Target information for TESS Data Alerts.

Returns

Pandas DataFrame of containing target information

6.42.4 Member Data Documentation**6.42.4.1 end_url**

```
skdaccess.astro.tess.data.cache.DataFetcher.end_url
```

6.42.4.2 start_url

```
skdaccess.astro.tess.data.cache.DataFetcher.start_url
```

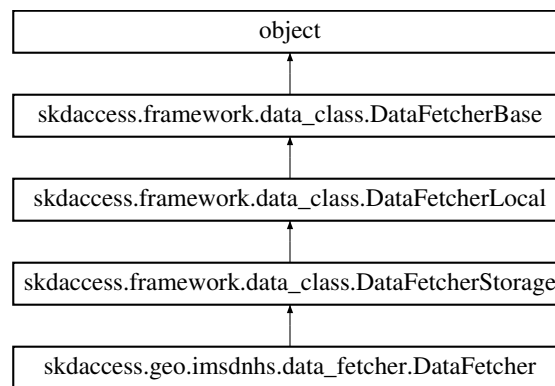
The documentation for this class was generated from the following file:

- [astro/tess/data/cache.py](#)

6.43 skdaccess.geo.imsdnhs.DataFetcher Class Reference

Fetches data for the Interactive Multisensor Snow and Ice Mapping System Daily Northern Hemisphere Snow and Ice Analysis.

Inheritance diagram for skdaccess.geo.imsdnhs.DataFetcher:



Public Member Functions

- `def __init__(self, coordinate_dict, start_date, end_date)`
Initializes the Data Fetcher.
- `def output(self)`
Fetch snow coverage data for coordinates.
- `def downloadFullDataset(cls, out_file, use_file=None)`
Abstract function used to download full data set.
- `def multirun_enabled(self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getDataLocation(data_name)`
Get the location of data set.
- `def setDataLocation(data_name, location, key='data_location')`
Set the location of a data set.
- `def perturb(self)`
Perturb parameters.
- `def reset(self)`
Set all parameters to initial value.

- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- `coordinate_dict`
- `start_date`
- `end_date`
- `ap_paramList`
- `verbose`

6.43.1 Detailed Description

Fetches data for the Interactive Multisensor Snow and Ice Mapping System Daily Northern Hemisphere Snow and Ice Analysis.

6.43.2 Constructor & Destructor Documentation

6.43.2.1 `__init__`()

```
def skdaccess.geo.imsdnhs.DataFetcher.__init__ (
    self,
    coordinate_dict,
    start_date,
    end_date )
```

Initializes the Data Fetcher.

Parameters

<i>coordinate_dict</i>	Dictionary of locations where the names are the keys and the items are lists containing the latitude and longitude are the values
<i>start_date</i>	Starting date
<i>end_date</i>	Ending date

6.43.3 Member Function Documentation

6.43.3.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.43.3.2 downloadFullDataset()

```
def skdaccess.framework.data_class.DataFetcherStorage.downloadFullDataset (
    cls,
    out_file,
    use_file = None ) [inherited]
```

Abstract function used to download full data set.

Parameters

<i>out_file</i>	output file name
<i>use_file</i>	Use previously downloaded data

Returns

Absolute path of parsed data

6.43.3.3 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.43.3.4 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.43.3.5 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.43.3.6 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.43.3.7 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStorage.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.43.3.8 output()

```
def skdaccess.geo.imsdnhs.DataFetcher.output (
    self )
```

Fetch snow coverage data for coordinates.

Returns

Data wrapper for snow coverage

6.43.3.9 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.43.3.10 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.43.3.11 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.43.3.12 `verbose_print()`

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.43.3.13 `writeConfig()`

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.43.3.14 `writeConfigItem()`

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.43.4 Member Data Documentation**6.43.4.1 ap_paramList**

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.43.4.2 coordinate_dict

`skdaccess.geo.imsdnhs.DataFetcher.coordinate_dict`

6.43.4.3 end_date

`skdaccess.geo.imsdnhs.DataFetcher.end_date`

6.43.4.4 start_date

`skdaccess.geo.imsdnhs.DataFetcher.start_date`

6.43.4.5 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

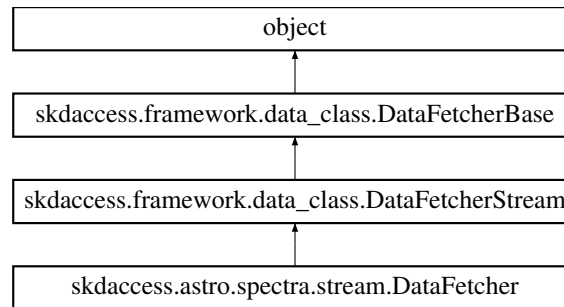
The documentation for this class was generated from the following file:

- [geo/imsdnhs/data_fetcher.py](#)

6.44 skdaccess.astro.spectra.stream.DataFetcher Class Reference

Data Fetcher for Sloan Digital Sky Survey spectra.

Inheritance diagram for skdaccess.astro.spectra.stream.DataFetcher:



Public Member Functions

- def `__init__` (self, `ap_paramList`)
Initialize SDSS spectra Data Fetcher.
- def `output` (self)
Generate data wrapper.
- def `retrieveOnlineData` (self, `data_specification`)
Method for downloading data into memory.
- def `multirun_enabled` (self)
Returns whether or not this data fetcher is multirun enabled.
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [ap_paramList](#)
- [verbose](#)

6.44.1 Detailed Description

Data Fetcher for Sloan Digital Sky Survey spectra.

6.44.2 Constructor & Destructor Documentation

6.44.2.1 `__init__()`

```
def skdaccess.astro.spectra.stream.DataFetcher.__init__ (
    self,
    ap_paramList )
```

Initialize SDSS spectra Data Fetcher.

Parameters

<code>ap_paramList[url_list]</code>	Autolist of URLs to access
-------------------------------------	----------------------------

6.44.3 Member Function Documentation

6.44.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.44.3.2 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.44.3.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.44.3.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.44.3.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.44.3.6 output()

```
def skdaccess.astro.spectra.stream.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

Table wrapper of SDSS spectra data

6.44.3.7 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.44.3.8 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.44.3.9 retrieveOnlineData()

```
def skdaccess.framework.data_class.DataFetcherStream.retrieveOnlineData (
    self,
    data_specification ) [inherited]
```

Method for downloading data into memory.

Parameters

<i>data_specification</i>	Url list of data to be retrieved
---------------------------	----------------------------------

Returns

Retrieved data

6.44.3.10 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.44.3.11 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.44.3.12 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
```

```

    key,
    value ) [inherited]

```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.44.4 Member Data Documentation

6.44.4.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]
```

6.44.4.2 verbose

```
skdaccess.framework.data_class.DataFetcherBase.verbose [inherited]
```

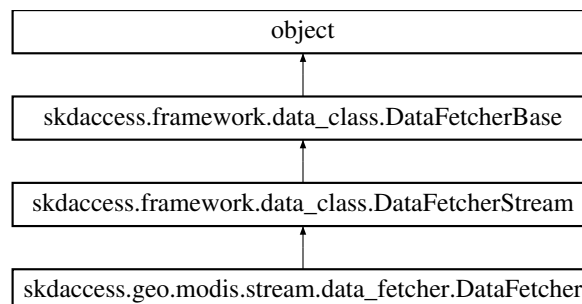
The documentation for this class was generated from the following file:

- astro/spectra/[stream.py](#)

6.45 skdaccess.geo.modis.stream.DataFetcher Class Reference

Data Fetcher for MODIS data.

Inheritance diagram for skdaccess.geo.modis.stream.DataFetcher:



Public Member Functions

- `def __init__ (self, ap_paramList, modis_platform, modis_id, variable_list, start_date, end_date, daynightboth='D', grid=None, grid_fill=np.nan, use_long_name=False)`
Construct Data Fetcher object.
- `def output (self)`
Generate data wrapper.
- `def retrieveOnlineData (self, data_specification)`
Method for downloading data into memory.
- `def multirun_enabled (self)`
Returns whether or not this data fetcher is multirun enabled.
- `def perturb (self)`
Perturb parameters.
- `def reset (self)`
Set all parameters to initial value.
- `def __str__ (self)`
Generate string description.
- `def getMetadata (self)`
Return metadata about Data Fetcher.
- `def getConfig ()`
Retrieve skdaccess configuration.
- `def getConfigItem (section, key)`
Retrieve skdaccess configuration item.
- `def writeConfigItem (section, key, value)`
Retrieve skdaccess configuration item.
- `def writeConfig (conf)`
Write config to disk.
- `def verbose_print (self, args, kwargs)`
Print statement if verbose flag is set.

Public Attributes

- `modis_id`
- `variable_list`
- `start_date`
- `end_date`
- `daynightboth`
- `grid`
- `grid_fill`
- `use_long_name`
- `modis_platform`
- `modis_identifier`
- `ap_paramList`
- `verbose`

6.45.1 Detailed Description

Data Fetcher for MODIS data.

6.45.2 Constructor & Destructor Documentation

6.45.2.1 __init__()

```
def skdaccess.geo.modis.stream.DataFetcher.__init__ (
    self,
    ap_paramList,
    modis_platform,
    modis_id,
    variable_list,
    start_date,
    end_date,
    daynightboth = 'D',
    grid = None,
    grid_fill = np.nan,
    use_long_name = False )
```

Construct Data Fetcher object.

Parameters

<i>ap_paramList[lat]</i>	Search latitude
<i>ap_paramList[lon]</i>	Search longitude
<i>modis_platform</i>	Platform (Either "Terra" or "Aqua")
<i>modis_id</i>	Product string (e.g. '06_L2')
<i>variable_list</i>	List of variables to fetch
<i>start_date</i>	Starting date
<i>end_date</i>	Ending date
<i>daynightboth</i>	Use daytime data ('D'), nighttime data ('N') or both ('B')
<i>grid</i>	Further divide each image into a multiple grids of size (y,x)
<i>grid_fill</i>	Fill value to use when creating gridded data
<i>use_long_name</i>	Use long names for metadata instead of variable name

6.45.3 Member Function Documentation

6.45.3.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.45.3.2 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.45.3.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.45.3.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.45.3.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.45.3.6 output()

```
def skdaccess.geo.modis.stream.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

data wrapper of MODIS data

6.45.3.7 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.45.3.8 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.45.3.9 retrieveOnlineData()

```
def skdaccess.framework.data_class.DataFetcherStream.retrieveOnlineData (
    self,
    data_specification ) [inherited]
```

Method for downloading data into memory.

Parameters

<i>data_specification</i>	Url list of data to be retrieved
---------------------------	----------------------------------

Returns

Retrieved data

6.45.3.10 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.45.3.11 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.45.3.12 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
```



```
key,  
value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.45.4 Member Data Documentation

6.45.4.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]
```

6.45.4.2 daynightboth

```
skdaccess.geo.modis.stream.DataFetcher.daynightboth
```

6.45.4.3 end_date

```
skdaccess.geo.modis.stream.DataFetcher.end_date
```

6.45.4.4 grid

```
skdaccess.geo.modis.stream.DataFetcher.grid
```

6.45.4.5 grid_fill

`skdaccess.geo.modis.stream.DataFetcher.grid_fill`

6.45.4.6 modis_id

`skdaccess.geo.modis.stream.DataFetcher.modis_id`

6.45.4.7 modis_identifier

`skdaccess.geo.modis.stream.DataFetcher.modis_identifier`

6.45.4.8 modis_platform

`skdaccess.geo.modis.stream.DataFetcher.modis_platform`

6.45.4.9 start_date

`skdaccess.geo.modis.stream.DataFetcher.start_date`

6.45.4.10 use_long_name

`skdaccess.geo.modis.stream.DataFetcher.use_long_name`

6.45.4.11 variable_list

`skdaccess.geo.modis.stream.DataFetcher.variable_list`

6.45.4.12 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

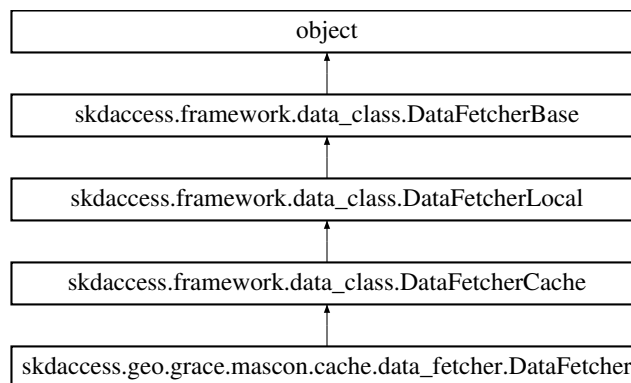
The documentation for this class was generated from the following file:

- [geo/modis/stream/data_fetcher.py](#)

6.46 skdaccess.geo.grace.mascon.cache.DataFetcher Class Reference

Data Fetcher for GRACE mascon data.

Inheritance diagram for `skdaccess.geo.grace.mascon.cache.DataFetcher`:



Public Member Functions

- `def __init__(self, ap_paramList, start_date=None, end_date=None)`
Construct a GRACE mascon Data Fetcher.
- `def output(self)`
Create a datawrapper containing GRACE mascon data.
- `def getMasconPlacement(self)`
Retrieve mascon placement data.
- `def checkIfDataExists(self, in_file_name)`
Checks if the file exists on the filesystem and the file is not empty.
- `def cacheData(self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)`
Download and store specified data to local disk.
- `def multirun_enabled(self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getHDFStorage(self, keyname)`
Retrieve a Pandas HDF Store for a dataset.
- `def getDataLocation(data_name)`
Get the location of data set.

- def `setDataLocation` (data_name, location, key='data_location')
Set the location of a data set.
- def `perturb` (self)
Perturb parameters.
- def `reset` (self)
Set all parameters to initial value.
- def `__str__` (self)
Generate string description.
- def `getMetadata` (self)
Return metadata about Data Fetcher.
- def `getConfig` ()
Retrieve skdaccess configuration.
- def `getConfigItem` (section, key)
Retrieve skdaccess configuration item.
- def `writeConfigItem` (section, key, value)
Retrieve skdaccess configuration item.
- def `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- `start_date`
- `end_date`
- `mascon_url`
- `scale_factor_url`
- `mascon_placement_url`
- `ap_paramList`
- `verbose`

6.46.1 Detailed Description

Data Fetcher for GRACE mascon data.

6.46.2 Constructor & Destructor Documentation

6.46.2.1 `__init__()`

```
def skdaccess.geo.grace.mascon.cache.DataFetcher.__init__ (
    self,
    ap_paramList,
    start_date = None,
    end_date = None )
```

Construct a GRACE mascon Data Fetcher.

Parameters

<i>ap_paramList[geo_point]</i>	AutoList of geographic location tuples (lat,lon)
<i>start_date</i>	Beginning date
<i>end_date</i>	Ending date

6.46.3 Member Function Documentation

6.46.3.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.46.3.2 cacheData()

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.46.3.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.46.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.46.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.46.3.6 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.46.3.7 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.46.3.8 getMasconPlacement()

```
def skdaccess.geo.grace.mascon.cache.DataFetcher.getMasconPlacement (
    self )
```

Retrieve mascon placement data.

Returns

Mascon data, Mascon metadata

6.46.3.9 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.46.3.10 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.46.3.11 output()

```
def skdaccess.geo.grace.mascon.cache.DataFetcher.output (
    self )
```

Create a datawrapper containing GRACE mascon data.

Returns

Table Datawrapper containing Mascon GRACE data

6.46.3.12 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.46.3.13 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.46.3.14 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.46.3.15 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.46.3.16 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.46.3.17 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.46.4 Member Data Documentation

6.46.4.1 ap_paramList

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.46.4.2 end_date

`skdaccess.geo.grace.mascon.cache.DataFetcher.end_date`

6.46.4.3 mascon_placement_url

`skdaccess.geo.grace.mascon.cache.DataFetcher.mascon_placement_url`

6.46.4.4 mascon_url

`skdaccess.geo.grace.mascon.cache.DataFetcher.mascon_url`

6.46.4.5 scale_factor_url

`skdaccess.geo.grace.mascon.cache.DataFetcher.scale_factor_url`

6.46.4.6 start_date

`skdaccess.geo.grace.mascon.cache.DataFetcher.start_date`

6.46.4.7 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

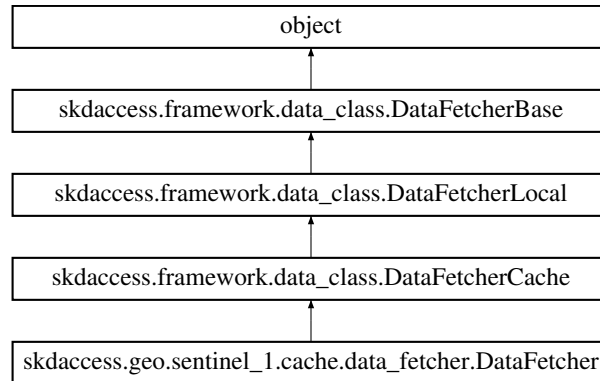
The documentation for this class was generated from the following file:

- `geo/grace/mascon/cache/data_fetcher.py`

6.47 skdaccess.geo.sentinel_1.cache.DataFetcher Class Reference

[DataFetcher](#) for retrieving Sentinel SLC data.

Inheritance diagram for skdaccess.geo.sentinel_1.cache.DataFetcher:



Public Member Functions

- `def __init__ (self, url_list, satellite_url_list, username, password, swath, polarization='VV', local_paths=False, verbose=True)`
Initialize Sentinel Data Fetcher.
- `def output (self)`
Generate data wrapper.
- `def checkIfDataExists (self, in_file_name)`
Checks if the file exists on the filesystem and the file is not empty.
- `def cacheData (self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)`
Download and store specified data to local disk.
- `def multirun_enabled (self)`
Returns whether or not this data fetcher is multirun enabled.
- `def getHDFStorage (self, keyname)`
Retrieve a Pandas HDF Store for a dataset.
- `def getDataLocation (data_name)`
Get the location of data set.
- `def setDataLocation (data_name, location, key='data_location')`
Set the location of a data set.
- `def perturb (self)`
Perturb parameters.
- `def reset (self)`
Set all parameters to initial value.
- `def __str__ (self)`
Generate string description.
- `def getMetadata (self)`
Return metadata about Data Fetcher.

- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [url_list](#)
- [satellite_url_list](#)
- [swath](#)
- [username](#)
- [password](#)
- [polarization](#)
- [local_paths](#)
- [ap_paramList](#)
- [verbose](#)

6.47.1 Detailed Description

[DataFetcher](#) for retrieving Sentinel SLC data.

6.47.2 Constructor & Destructor Documentation

6.47.2.1 `__init__()`

```
def skdaccess.geo.sentinel_1.cache.DataFetcher.__init__ (
    self,
    url_list,
    satellite_url_list,
    username,
    password,
    swath,
    polarization = 'VV',
    local_paths = False,
    verbose = True )
```

Initialize Sentinel Data Fetcher.

Parameters

<i>url_list</i>	List of urls of SLC data
<i>satellite_url_list</i>	List of satellite urls
<i>username</i>	Username for downloading data
<i>password</i>	Password for downloading data
<i>swath</i>	Swath number (1, 2, or 3)
<i>polarization</i>	Polarization of data to retrieve
<i>local_paths</i>	locations are local paths, not urls
<i>verbose</i>	Print additional information

6.47.3 Member Function Documentation

6.47.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.47.3.2 `cacheData()`

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True ) [inherited]
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.47.3.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name ) [inherited]
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.47.3.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.47.3.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.47.3.6 `getDataLocation()`

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.47.3.7 `getHDFStorage()`

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname ) [inherited]
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.47.3.8 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.47.3.9 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.47.3.10 output()

```
def skdaccess.geo.sentinel_1.cache.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

Sentinel SLC data in a data wrapper

6.47.3.11 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.47.3.12 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.47.3.13 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.47.3.14 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.47.3.15 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.47.3.16 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.47.4 Member Data Documentation

6.47.4.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]
```

6.47.4.2 local_paths

```
skdaccess.geo.sentinel_1.cache.DataFetcher.local_paths
```

6.47.4.3 password

`skdaccess.geo.sentinel_1.cache.DataFetcher.password`

6.47.4.4 polarization

`skdaccess.geo.sentinel_1.cache.DataFetcher.polarization`

6.47.4.5 satellite_url_list

`skdaccess.geo.sentinel_1.cache.DataFetcher.satellite_url_list`

6.47.4.6 swath

`skdaccess.geo.sentinel_1.cache.DataFetcher.swath`

6.47.4.7 url_list

`skdaccess.geo.sentinel_1.cache.DataFetcher.url_list`

6.47.4.8 username

`skdaccess.geo.sentinel_1.cache.DataFetcher.username`

6.47.4.9 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

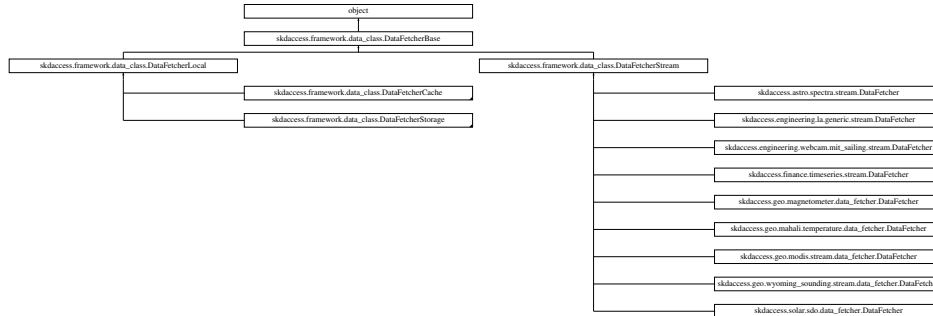
The documentation for this class was generated from the following file:

- `geo/sentinel_1/cache/data_fetcher.py`

6.48 skdaccess.framework.data_class.DataFetcherBase Class Reference

Base class for all data fetchers.

Inheritance diagram for skdaccess.framework.data_class.DataFetcherBase:



Public Member Functions

- `def __init__ (self, ap_paramList=[], verbose=False)`
Initialize data fetcher with parameter list.
- `def output (self)`
Output data wrapper.
- `def perturb (self)`
Perturb parameters.
- `def reset (self)`
Set all parameters to initial value.
- `def __str__ (self)`
Generate string description.
- `def getMetadata (self)`
Return metadata about Data Fetcher.
- `def getConfig ()`
Retrieve skdaccess configuration.
- `def getConfigItem (section, key)`
Retrieve skdaccess configuration item.
- `def writeConfigItem (section, key, value)`
Retrieve skdaccess configuration item.
- `def writeConfig (conf)`
Write config to disk.
- `def multirun_enabled (self)`
Returns whether or not this data fetcher is multirun enabled.
- `def verbose_print (self, args, kwargs)`
Print statement if verbose flag is set.

Public Attributes

- `ap_paramList`
- `verbose`

6.48.1 Detailed Description

Base class for all data fetchers.

6.48.2 Constructor & Destructor Documentation

6.48.2.1 `__init__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__init__ (
    self,
    ap_paramList = [],
    verbose = False )
```

Initialize data fetcher with parameter list.

Parameters

<i>ap_paramList</i>	List of parameters
<i>verbose</i>	Output extra information

6.48.3 Member Function Documentation

6.48.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self )
```

Generate string description.

6.48.3.2 `getConfig()`

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( )
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.48.3.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key )
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.48.3.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self )
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.48.3.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherBase.multirun_enabled (
    self )
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.48.3.6 output()

```
def skdaccess.framework.data_class.DataFetcherBase.output (
    self )
```

Output data wrapper.

Returns

Datawrapper

6.48.3.7 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self )
```

Perturb parameters.

6.48.3.8 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self )
```

Set all parameters to initial value.

6.48.3.9 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs )
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.48.3.10 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf )
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.48.3.11 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value )
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.48.4 Member Data Documentation

6.48.4.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList
```

6.48.4.2 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose`

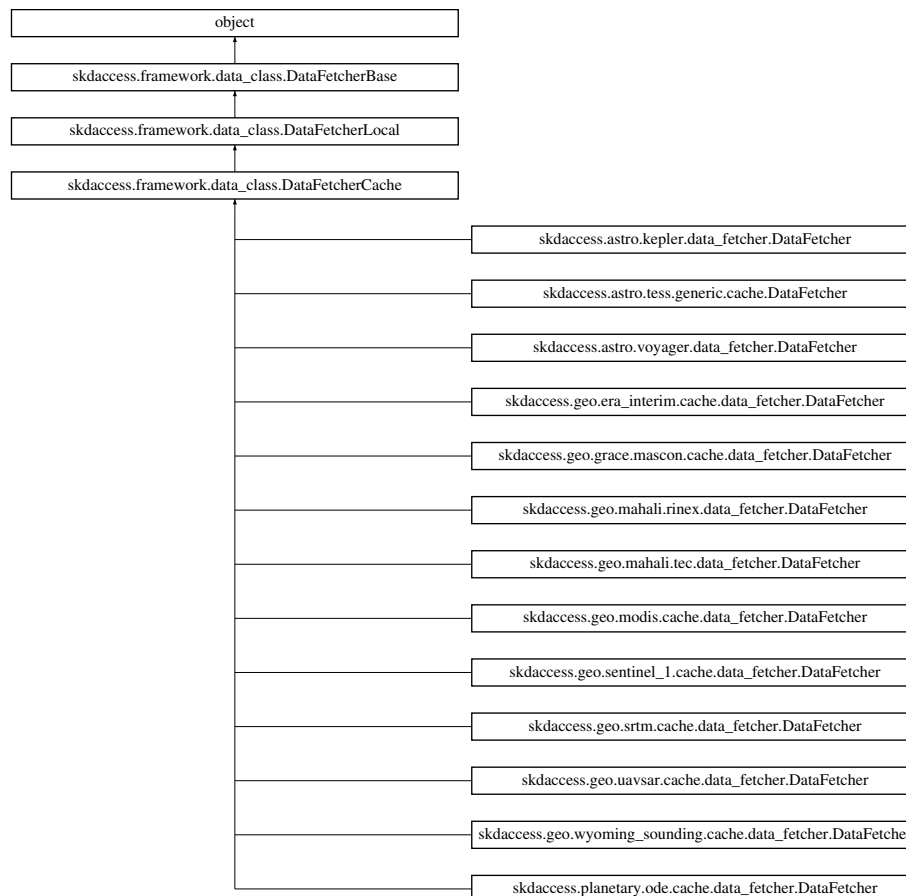
The documentation for this class was generated from the following file:

- [framework/data_class.py](#)

6.49 skdaccess.framework.data_class.DataFetcherCache Class Reference

Data fetcher base class for downloading data and caching results on hard disk.

Inheritance diagram for `skdaccess.framework.data_class.DataFetcherCache`:



Public Member Functions

- def [checkIfDataExists](#) (self, in_file_name)
Checks if the file exists on the filesystem and the file is not empty.
- def [cacheData](#) (self, keyname, online_path_list, username=None, password=None, authentication_url=None, cookiejar=None, use_requests=False, use_progress_bar=True)
Download and store specified data to local disk.
- def [multirun_enabled](#) (self)
Returns whether or not this data fetcher is multirun enabled.
- def [getHDFSStorage](#) (self, keyname)
Retrieve a Pandas HDF Store for a dataset.
- def [getDataLocation](#) (data_name)
Get the location of data set.
- def [setDataLocation](#) (data_name, location, key='data_location')
Set the location of a data set.
- def [output](#) (self)
Output data wrapper.
- def [perturb](#) (self)
Perturb parameters.
- def [reset](#) (self)
Set all parameters to initial value.
- def [__str__](#) (self)
Generate string description.
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [ap_paramList](#)
- [verbose](#)

6.49.1 Detailed Description

Data fetcher base class for downloading data and caching results on hard disk.

6.49.2 Member Function Documentation

6.49.2.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.49.2.2 `cacheData()`

```
def skdaccess.framework.data_class.DataFetcherCache.cacheData (
    self,
    keyname,
    online_path_list,
    username = None,
    password = None,
    authentication_url = None,
    cookiejar = None,
    use_requests = False,
    use_progress_bar = True )
```

Download and store specified data to local disk.

Parameters

<i>keyname</i>	Name of dataset in configuration file
<i>online_path_list</i>	List of urls to data
<i>username</i>	Username for accessing online resources
<i>password</i>	Password for accessing online resources
<i>authentication_url</i>	The url used for authentication (unused when use_requests=True)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when use_requests=True)
<i>use_requests</i>	Use the requests library instead of the standard library for accessing resources
<i>use_progress_bar</i>	Use a progress bar to show number of items downloaded

Returns

List of downloaded file locations

6.49.2.3 checkIfDataExists()

```
def skdaccess.framework.data_class.DataFetcherCache.checkIfDataExists (
    self,
    in_file_name )
```

Checks if the file exists on the filesystem and the file is not empty.

Parameters

<i>in_file_name</i>	Input filename to test
---------------------	------------------------

Returns

True if data exists and False otherwise

6.49.2.4 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.49.2.5 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.49.2.6 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.49.2.7 getHDFStorage()

```
def skdaccess.framework.data_class.DataFetcherCache.getHDFStorage (
    self,
    keyname )
```

Retrieve a Pandas HDF Store for a dataset.

Parameters

<i>keyname</i>	Key name of HDF store
----------------	-----------------------

Returns

Pandas HDF Store

6.49.2.8 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.49.2.9 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherCache.multirun_enabled (
    self )
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.49.2.10 output()

```
def skdaccess.framework.data_class.DataFetcherBase.output (
    self ) [inherited]
```

Output data wrapper.

Returns

Datawrapper

6.49.2.11 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.49.2.12 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.49.2.13 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.49.2.14 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.49.2.15 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.49.2.16 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.49.3 Member Data Documentation

6.49.3.1 ap_paramList

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.49.3.2 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

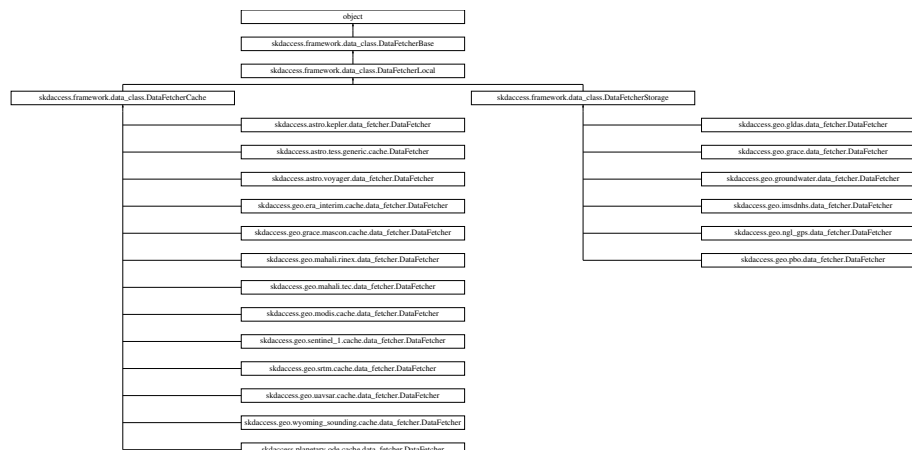
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.50 skdaccess.framework.data_class.DataFetcherLocal Class Reference

Data fetcher base class for use when storing data locally.

Inheritance diagram for `skdaccess.framework.data_class.DataFetcherLocal`:



Public Member Functions

- def [getDataLocation](#) (data_name)
Get the location of data set.
- def [setDataLocation](#) (data_name, location, key='data_location')
Set the location of a data set.
- def [output](#) (self)
Output data wrapper.
- def [perturb](#) (self)
Perturb parameters.
- def [reset](#) (self)
Set all parameters to initial value.
- def [__str__](#) (self)
Generate string description.
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [multirun_enabled](#) (self)
Returns whether or not this data fetcher is multirun enabled.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [ap_paramList](#)
- [verbose](#)

6.50.1 Detailed Description

Data fetcher base class for use when storing data locally.

6.50.2 Member Function Documentation

6.50.2.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.50.2.2 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.50.2.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.50.2.4 getDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name )
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.50.2.5 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.50.2.6 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherBase.multirun_enabled (
    self ) [inherited]
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.50.2.7 output()

```
def skdaccess.framework.data_class.DataFetcherBase.output (
    self ) [inherited]
```

Output data wrapper.

Returns

Datawrapper

6.50.2.8 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.50.2.9 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.50.2.10 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' )
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.50.2.11 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.50.2.12 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.50.2.13 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.50.3 Member Data Documentation

6.50.3.1 ap_paramList

skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]

6.50.3.2 verbose

skdaccess.framework.data_class.DataFetcherBase.verbose [inherited]

The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.51 skdaccess.framework.data_class.DataFetcherStorage Class Reference

Data fetcher base class for use when entire data set is downloaded.

Inheritance diagram for skdaccess.framework.data_class.DataFetcherStorage:



Public Member Functions

- def [downloadFullDataset](#) (cls, out_file, use_file=None)
Abstract function used to download full data set.
- def [multirun_enabled](#) (self)
Returns whether or not this data fetcher is multirun enabled.
- def [getDataLocation](#) (data_name)
Get the location of data set.
- def [setDataLocation](#) (data_name, location, key='data_location')
Set the location of a data set.
- def [output](#) (self)
Output data wrapper.
- def [perturb](#) (self)
Perturb parameters.
- def [reset](#) (self)
Set all parameters to initial value.
- def [__str__](#) (self)
Generate string description.
- def [getMetadata](#) (self)

- Return metadata about Data Fetcher.*
 - def `getConfig` ()
- Retrieve skdaccess configuration.*
 - def `getConfigItem` (section, key)
- Retrieve skdaccess configuration item.*
 - def `writeConfigItem` (section, key, value)
- Retrieve skdaccess configuration item.*
 - def `writeConfig` (conf)
- Write config to disk.*
 - def `verbose_print` (self, args, kwargs)
- Print statement if verbose flag is set.*

Public Attributes

- `ap_paramList`
- `verbose`

6.51.1 Detailed Description

Data fetcher base class for use when entire data set is downloaded.

6.51.2 Member Function Documentation

6.51.2.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.51.2.2 `downloadFullDataset()`

```
def skdaccess.framework.data_class.DataFetcherStorage.downloadFullDataset (
    cls,
    out_file,
    use_file = None )
```

Abstract function used to download full data set.

Parameters

<i>out_file</i>	output file name
<i>use_file</i>	Use previously downloaded data

Returns

Absolute path of parsed data

6.51.2.3 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.51.2.4 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.51.2.5 `getDataLocation()`

```
def skdaccess.framework.data_class.DataFetcherLocal.getDataLocation (
    data_name ) [inherited]
```

Get the location of data set.

Parameters

<i>data_name</i>	Name of data set
------------------	------------------

Returns

string of data location, None if not found

6.51.2.6 `getMetadata()`

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.51.2.7 `multirun_enabled()`

```
def skdaccess.framework.data_class.DataFetcherStorage.multirun_enabled (
    self )
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.51.2.8 output()

```
def skdaccess.framework.data_class.DataFetcherBase.output (
    self ) [inherited]
```

Output data wrapper.

Returns

Datawrapper

6.51.2.9 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.51.2.10 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.51.2.11 setDataLocation()

```
def skdaccess.framework.data_class.DataFetcherLocal.setDataLocation (
    data_name,
    location,
    key = 'data_location' ) [inherited]
```

Set the location of a data set.

Parameters

<i>data_name</i>	Name of data set
<i>location</i>	Location of data set
<i>key</i>	Key of configuration option

6.51.2.12 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.51.2.13 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.51.2.14 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
    key,
    value ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.51.3 Member Data Documentation

6.51.3.1 ap_paramList

`skdaccess.framework.data_class.DataFetcherBase.ap_paramList` [inherited]

6.51.3.2 verbose

`skdaccess.framework.data_class.DataFetcherBase.verbose` [inherited]

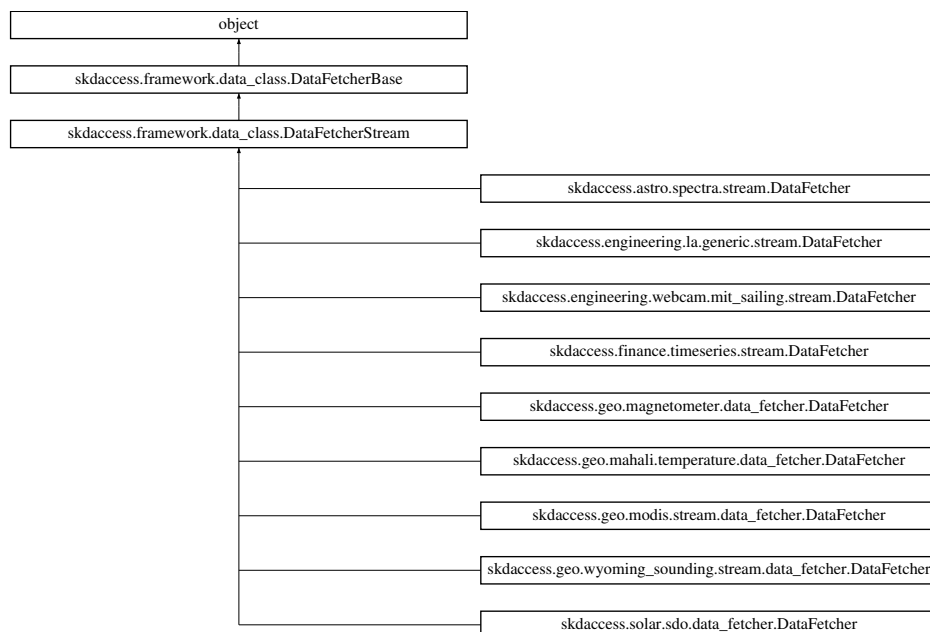
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.52 skdaccess.framework.data_class.DataFetcherStream Class Reference

Data fetcher base class for downloading data into memory.

Inheritance diagram for `skdaccess.framework.data_class.DataFetcherStream`:



Public Member Functions

- def [retrieveOnlineData](#) (self, data_specification)
Method for downloading data into memory.
- def [multirun_enabled](#) (self)
Returns whether or not this data fetcher is multirun enabled.
- def [output](#) (self)
Output data wrapper.
- def [perturb](#) (self)
Perturb parameters.
- def [reset](#) (self)
Set all parameters to initial value.
- def [__str__](#) (self)
Generate string description.
- def [getMetadata](#) (self)
Return metadata about Data Fetcher.
- def [getConfig](#) ()
Retrieve skdaccess configuration.
- def [getConfigItem](#) (section, key)
Retrieve skdaccess configuration item.
- def [writeConfigItem](#) (section, key, value)
Retrieve skdaccess configuration item.
- def [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- [ap_paramList](#)
- [verbose](#)

6.52.1 Detailed Description

Data fetcher base class for downloading data into memory.

6.52.2 Member Function Documentation

6.52.2.1 __str__()

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self ) [inherited]
```

Generate string description.

6.52.2.2 getConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( ) [inherited]
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.52.2.3 getConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.getConfigItem (
    section,
    key ) [inherited]
```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value

Returns

Requested configuration item or None if it doesn't exist

6.52.2.4 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self ) [inherited]
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.52.2.5 multirun_enabled()

```
def skdaccess.framework.data_class.DataFetcherStream.multirun_enabled (
    self )
```

Returns whether or not this data fetcher is multirun enabled.

Returns

Boolean indicating whether or not this data fetcher is multirun enabled

6.52.2.6 output()

```
def skdaccess.framework.data_class.DataFetcherBase.output (
    self ) [inherited]
```

Output data wrapper.

Returns

Datawrapper

6.52.2.7 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self ) [inherited]
```

Perturb parameters.

6.52.2.8 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self ) [inherited]
```

Set all parameters to initial value.

6.52.2.9 retrieveOnlineData()

```
def skdaccess.framework.data_class.DataFetcherStream.retrieveOnlineData (
    self,
    data_specification )
```

Method for downloading data into memory.

Parameters

<i>data_specification</i>	Url list of data to be retrieved
---------------------------	----------------------------------

Returns

Retrieved data

6.52.2.10 verbose_print()

```
def skdaccess.framework.data_class.DataFetcherBase.verbose_print (
    self,
    args,
    kwargs ) [inherited]
```

Print statement if verbose flag is set.

Parameters

<i>*args</i>	Arguments to pass to print
<i>**kwargs</i>	Keyword arguments to pass to print

6.52.2.11 writeConfig()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfig (
    conf ) [inherited]
```

Write config to disk.

Parameters

<i>conf</i>	configparser.ConfigParser object
-------------	----------------------------------

6.52.2.12 writeConfigItem()

```
def skdaccess.framework.data_class.DataFetcherBase.writeConfigItem (
    section,
```

```

        key,
        value ) [inherited]

```

Retrieve skdaccess configuration item.

Parameters

<i>section</i>	Section of configuration item
<i>key</i>	Configuration key value
<i>value</i>	Value to be written

Returns

Requested configuration item or None if it doesn't exist

6.52.3 Member Data Documentation

6.52.3.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList [inherited]
```

6.52.3.2 verbose

```
skdaccess.framework.data_class.DataFetcherBase.verbose [inherited]
```

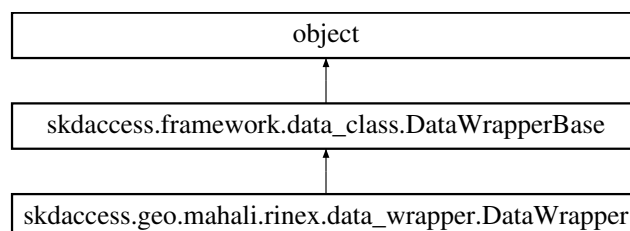
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.53 skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper Class Reference

Data wrapper for Mahali data.

Inheritance diagram for skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper:



Public Member Functions

- def [getIterator](#) (self)
Get iterator to Mahali data.
- def [update](#) (self, obj)
Updated wrapped data.
- def [updateMetadata](#) (self, new_metadata)
Update metadata.
- def [get](#) (self)
Retrieve stored data.
- def [getResults](#) (self)
Retrieve accumulated results, if any.
- def [addResult](#) (self, rkey, rres)
Add a result to the data wrapper.
- def [reset](#) (self)
Reset data back to original state.
- def [info](#) (self, key=None)
Get information about data wrapper.
- def [__len__](#) (self)
Get length of wrapped data.
- def [getRunID](#) (self)
Get the Run ID.

Public Attributes

- [data](#)
- [results](#)
- [constants](#)
- [run_id](#)
- [meta_data](#)

6.53.1 Detailed Description

Data wrapper for Mahali data.

6.53.2 Member Function Documentation

6.53.2.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.53.2.2 `addResult()`

```
def skdaccess.framework.data_class.DataWrapperBase.addResult (
    self,
    rkey,
    rres ) [inherited]
```

Add a result to the data wrapper.

Parameters

<i>rkey</i>	Result key
<i>rres</i>	Result

6.53.2.3 `get()`

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.53.2.4 getIterator()

```
def skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper.getIterator (
    self )
```

Get iterator to Mahali data.

Returns

Iterator yielding (site,date,nav,obs)

6.53.2.5 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.53.2.6 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.53.2.7 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.53.2.8 reset()

```
def skdaccess.framework.data_class.DataWrapperBase.reset (
    self ) [inherited]
```

Reset data back to original state.

6.53.2.9 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.53.2.10 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.53.3 Member Data Documentation

6.53.3.1 constants

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.53.3.2 data

`skdaccess.framework.data_class.DataWrapperBase.data` [inherited]

6.53.3.3 meta_data

`skdaccess.framework.data_class.DataWrapperBase.meta_data` [inherited]

6.53.3.4 results

`skdaccess.framework.data_class.DataWrapperBase.results` [inherited]

6.53.3.5 run_id

`skdaccess.framework.data_class.DataWrapperBase.run_id` [inherited]

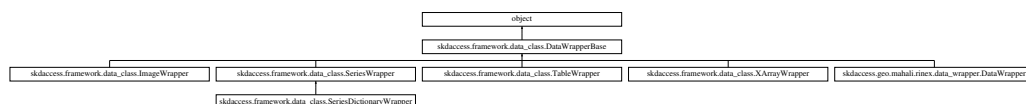
The documentation for this class was generated from the following file:

- [geo/mahali/rinex/data_wrapper.py](#)

6.54 skdaccess.framework.data_class.DataWrapperBase Class Reference

Base class for wrapping data for use in DiscoveryPipeline.

Inheritance diagram for `skdaccess.framework.data_class.DataWrapperBase`:



Public Member Functions

- def `__init__` (self, obj_wrap, run_id=-1, meta_data=None)
Construct wrapper from input data.
- def `update` (self, obj)
Updated wrapped data.
- def `updateMetadata` (self, new_metadata)
Update metadata.
- def `get` (self)
Retrieve stored data.
- def `getResults` (self)
Retrieve accumulated results, if any.
- def `addResult` (self, rkey, rres)
Add a result to the data wrapper.
- def `reset` (self)
Reset data back to original state.
- def `info` (self, key=None)
Get information about data wrapper.
- def `getIterator` (self)
Get an iterator to the data.
- def `__len__` (self)
Get length of wrapped data.
- def `getRunID` (self)
Get the Run ID.

Public Attributes

- `data`
- `results`
- `constants`
- `run_id`
- `meta_data`

6.54.1 Detailed Description

Base class for wrapping data for use in DiscoveryPipeline.

6.54.2 Constructor & Destructor Documentation

6.54.2.1 `__init__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__init__ (
    self,
    obj_wrap,
    run_id = -1,
    meta_data = None )
```

Construct wrapper from input data.

Parameters

<i>obj_wrap</i>	Data to be wrapped
<i>run_id</i>	ID of the run
<i>meta_data</i>	Metadata to store with data

6.54.3 Member Function Documentation

6.54.3.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self )
```

Get length of wrapped data.

Returns

length of wrapped data

6.54.3.2 `addResult()`

```
def skdaccess.framework.data_class.DataWrapperBase.addResult (
    self,
    rkey,
    rres )
```

Add a result to the data wrapper.

Parameters

<i>rkey</i>	Result key
<i>rres</i>	Result

6.54.3.3 `get()`

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self )
```

Retrieve stored data.

Returns

Stored data

6.54.3.4 getIterator()

```
def skdaccess.framework.data_class.DataWrapperBase.getIterator (
    self )
```

Get an iterator to the data.

Returns

iterator to data

6.54.3.5 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self )
```

Retrieve accumulated results, if any.

Returns

store results

6.54.3.6 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self )
```

Get the Run ID.

Returns

run_id

6.54.3.7 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None )
```

Get information about data wrapper.

Returns

The stored metadata

6.54.3.8 reset()

```
def skdaccess.framework.data_class.DataWrapperBase.reset (
    self )
```

Reset data back to original state.

6.54.3.9 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj )
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.54.3.10 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata )
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.54.4 Member Data Documentation**6.54.4.1 constants**

`skdaccess.framework.data_class.DataWrapperBase.constants`

6.54.4.2 data

`skdaccess.framework.data_class.DataWrapperBase.data`

6.54.4.3 meta_data

`skdaccess.framework.data_class.DataWrapperBase.meta_data`

6.54.4.4 results

`skdaccess.framework.data_class.DataWrapperBase.results`

6.54.4.5 run_id

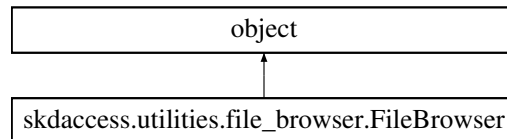
`skdaccess.framework.data_class.DataWrapperBase.run_id`

The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.55 skdaccess.utilities.file_browser.FileBrowser Class Reference

Inheritance diagram for skdaccess.utilities.file_browser.FileBrowser:



Public Member Functions

- def `__init__`(self)
- def `widget`(self)

Public Attributes

- `path`
- `files`
- `dirs`

6.55.1 Constructor & Destructor Documentation

6.55.1.1 `__init__()`

```
def skdaccess.utilities.file_browser.FileBrowser.__init__ (
    self )
```

6.55.2 Member Function Documentation

6.55.2.1 `widget()`

```
def skdaccess.utilities.file_browser.FileBrowser.widget (
    self )
```

6.55.3 Member Data Documentation

6.55.3.1 dirs

`skdaccess.utilities.file_browser.FileBrowser.dirs`

6.55.3.2 files

`skdaccess.utilities.file_browser.FileBrowser.files`

6.55.3.3 path

`skdaccess.utilities.file_browser.FileBrowser.path`

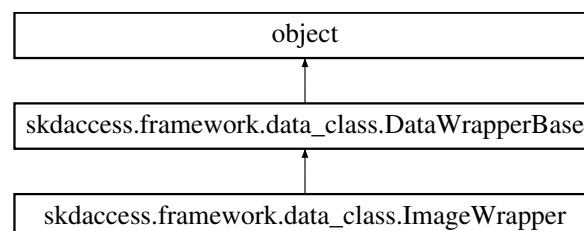
The documentation for this class was generated from the following file:

- [utilities/file_browser.py](#)

6.56 skdaccess.framework.data_class.ImageWrapper Class Reference

Wrapper for image data.

Inheritance diagram for `skdaccess.framework.data_class.ImageWrapper`:



Public Member Functions

- def [getIterator](#) (self)
Get an iterator to the data.
- def [updateData](#) (self, label, new_data)
Change image.
- def [deleteData](#) (self, label)
Delete image.
- def [update](#) (self, obj)
Updated wrapped data.
- def [updateMetadata](#) (self, new_metadata)
Update metadata.
- def [get](#) (self)
Retrieve stored data.
- def [getResults](#) (self)
Retrieve accumulated results, if any.
- def [addResult](#) (self, rkey, rres)
Add a result to the data wrapper.
- def [reset](#) (self)
Reset data back to original state.
- def [info](#) (self, key=None)
Get information about data wrapper.
- def [__len__](#) (self)
Get length of wrapped data.
- def [getRunID](#) (self)
Get the Run ID.

Public Attributes

- [data](#)
- [results](#)
- [constants](#)
- [run_id](#)
- [meta_data](#)

6.56.1 Detailed Description

Wrapper for image data.

6.56.2 Member Function Documentation

6.56.2.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.56.2.2 `addResult()`

```
def skdaccess.framework.data_class.DataWrapperBase.addResult (
    self,
    rkey,
    rres ) [inherited]
```

Add a result to the data wrapper.

Parameters

<i>rkey</i>	Result key
<i>rres</i>	Result

6.56.2.3 `deleteData()`

```
def skdaccess.framework.data_class.ImageWrapper.deleteData (
    self,
    label )
```

Delete image.

Parameters

<i>label</i>	Delete image with label
--------------	-------------------------

6.56.2.4 `get()`

```
def skdaccess.framework.data_class.DataWrapperBase.get (
```



```
self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.56.2.5 getIterator()

```
def skdaccess.framework.data_class.ImageWrapper.getIterator (
    self )
```

Get an iterator to the data.

Returns

Iterator yielding (label, image_data)

6.56.2.6 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.56.2.7 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.56.2.8 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.56.2.9 reset()

```
def skdaccess.framework.data_class.DataWrapperBase.reset (
    self ) [inherited]
```

Reset data back to original state.

6.56.2.10 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.56.2.11 updateData()

```
def skdaccess.framework.data_class.ImageWrapper.updateData (
    self,
    label,
    new_data )
```

Change image.

Parameters

<i>label</i>	Label of data to be changed
<i>new_data</i>	New data to replace old data

6.56.2.12 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.56.3 Member Data Documentation

6.56.3.1 constants

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.56.3.2 data

```
skdaccess.framework.data_class.DataWrapperBase.data [inherited]
```

6.56.3.3 meta_data

```
skdaccess.framework.data_class.DataWrapperBase.meta_data [inherited]
```

6.56.3.4 results

`skdaccess.framework.data_class.DataWrapperBase.results` [inherited]

6.56.3.5 run_id

`skdaccess.framework.data_class.DataWrapperBase.run_id` [inherited]

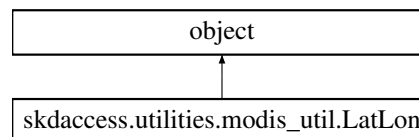
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.57 skdaccess.utilities.modis_util.LatLon Class Reference

Calculates Lat/Lon position from y,x pixel coordinate.

Inheritance diagram for `skdaccess.utilities.modis_util.LatLon`:



Public Member Functions

- `def __init__(self, metadata, x_offset=0, y_offset=0)`
Initialize getLatLon object.
- `def __call__(self, y, x)`
Convert pixel coordinates to lat/lon.

Public Attributes

- [x_offset](#)
- [y_offset](#)
- [lat_data](#)
- [lon_data](#)
- [alat](#)
- [alon](#)

6.57.1 Detailed Description

Calculates Lat/Lon position from y,x pixel coordinate.

6.57.2 Constructor & Destructor Documentation

6.57.2.1 __init__()

```
def skdaccess.utilities.modis_util.LatLon.__init__ (
    self,
    metadata,
    x_offset = 0,
    y_offset = 0 )
```

Initialize getLatLon object.

Parameters

<i>metadata</i>	Image metadata
<i>x_offset</i>	Pixel offset (used when gridding data)
<i>y_offset</i>	Pixel offset (used when gridding data)

6.57.3 Member Function Documentation

6.57.3.1 __call__()

```
def skdaccess.utilities.modis_util.LatLon.__call__ (
    self,
    y,
    x )
```

Convert pixel coordinates to lat/lon.

Parameters

<i>y</i>	y coordinate
<i>x</i>	x coordinate

Returns

(lat, lon)

6.57.4 Member Data Documentation**6.57.4.1 alat**

`skdaccess.utilities.modis_util.LatLon.alat`

6.57.4.2 alon

`skdaccess.utilities.modis_util.LatLon.alon`

6.57.4.3 lat_data

`skdaccess.utilities.modis_util.LatLon.lat_data`

6.57.4.4 lon_data

`skdaccess.utilities.modis_util.LatLon.lon_data`

6.57.4.5 x_offset

`skdaccess.utilities.modis_util.LatLon.x_offset`

6.57.4.6 y_offset

`skdaccess.utilities.modis_util.LatLon.y_offset`

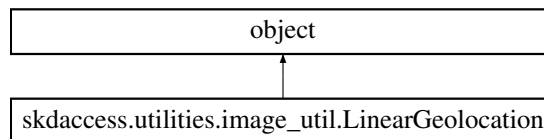
The documentation for this class was generated from the following file:

- [utilities/modis_util.py](#)

6.58 skdaccess.utilities.image_util.LinearGeolocation Class Reference

This class provides functions to convert between pixel and geodetic coordinates.

Inheritance diagram for skdaccess.utilities.image_util.LinearGeolocation:



Public Member Functions

- def `__init__` (self, data, extents, `x_offset=0`, `y_offset=0`, `flip_y=False`)
Initialize Linear Geolocation object.
- def `getLatLon` (self, y, x)
Retrive the latitude and longitude from pixel coordinates.
- def `getYX` (self, lat, lon)
Retrive the pixel coordinates from the latitude and longitude.
- def `getExtents` (self)
Retrieve the extents of the data.

Public Attributes

- `flip_y`
- `lon_extents`
- `lat_extents`
- `lat_pixel_size`
- `lon_pixel_size`
- `start_lat`
- `start_lon`
- `x_offset`
- `y_offset`
- `len_x`
- `len_y`

6.58.1 Detailed Description

This class provides functions to convert between pixel and geodetic coordinates.

Assumes a linear relationship between pixel and geodetic coordinates

6.58.2 Constructor & Destructor Documentation

6.58.2.1 `__init__()`

```
def skdaccess.utilities.image_util.LinearGeolocation.__init__ (
    self,
    data,
    extents,
    x_offset = 0,
    y_offset = 0,
    flip_y = False )
```

Initialize Linear Geolocation object.

Parameters

<i>data</i>	Numpy 2d data
<i>extents</i>	Latitude and longitude extents
<i>x_offset</i>	Pixel offset in x
<i>y_offset</i>	Pixel offset in y
<i>flip_y</i>	The y axis has been flipped so that increasing y values are decreasing in latitude

6.58.3 Member Function Documentation

6.58.3.1 `getExtents()`

```
def skdaccess.utilities.image_util.LinearGeolocation.getExtents (
    self )
```

Retrieve the extents of the data.

Returns

(minimum_longitude, maximum_longitude, minimum_latitude, maximum_latitude)

6.58.3.2 `getLatLon()`

```
def skdaccess.utilities.image_util.LinearGeolocation.getLatLon (
    self,
    y,
    x )
```

Retrive the latitude and longitude from pixel coordinates.

Parameters

<i>y</i>	The y pixel
<i>x</i>	The x pixel

Returns

(latitude, longitude) of the pixel coordinate

6.58.3.3 getYX()

```
def skdaccess.utilities.image_util.LinearGeolocation.getYX (
    self,
    lat,
    lon )
```

Retrive the pixel coordinates from the latitude and longitude.

Parameters

<i>lat</i>	The Latitude
<i>lon</i>	The Longitude

Returns

(y, x) pixel coordinates of the input latitude and longitude

6.58.4 Member Data Documentation**6.58.4.1 flip_y**

```
skdaccess.utilities.image_util.LinearGeolocation.flip_y
```

6.58.4.2 lat_extents

```
skdaccess.utilities.image_util.LinearGeolocation.lat_extents
```

6.58.4.3 lat_pixel_size

`skdaccess.utilities.image_util.LinearGeolocation.lat_pixel_size`

6.58.4.4 len_x

`skdaccess.utilities.image_util.LinearGeolocation.len_x`

6.58.4.5 len_y

`skdaccess.utilities.image_util.LinearGeolocation.len_y`

6.58.4.6 lon_extents

`skdaccess.utilities.image_util.LinearGeolocation.lon_extents`

6.58.4.7 lon_pixel_size

`skdaccess.utilities.image_util.LinearGeolocation.lon_pixel_size`

6.58.4.8 start_lat

`skdaccess.utilities.image_util.LinearGeolocation.start_lat`

6.58.4.9 start_lon

`skdaccess.utilities.image_util.LinearGeolocation.start_lon`

6.58.4.10 x_offset

```
skdaccess.utilities.image_util.LinearGeolocation.x_offset
```

6.58.4.11 y_offset

```
skdaccess.utilities.image_util.LinearGeolocation.y_offset
```

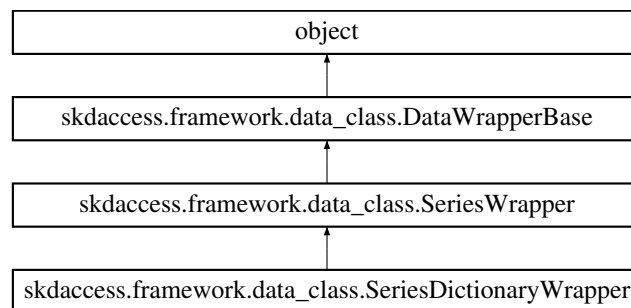
The documentation for this class was generated from the following file:

- [utilities/image_util.py](#)

6.59 skdaccess.framework.data_class.SeriesDictionaryWrapper Class Reference

Data wrapper for series data using a dictionary of data frames.

Inheritance diagram for skdaccess.framework.data_class.SeriesDictionaryWrapper:



Public Member Functions

- def [getIterator](#) (self)
Get an iterator to the data.
- def [getIndices](#) (self)
Get the indices of the data.
- def [getLength](#) (self)
Get total number of series that the iterate will loop over.
- def [update](#) (self, obj)
Updated wrapped data.
- def [updateMetadata](#) (self, new_metadata)
Update metadata.
- def [get](#) (self)
Retrieve stored data.

- def `getResults` (self)
Retrieve accumulated results, if any.
- def `addResult` (self, rkey, rres)
Add a result to the data wrapper.
- def `reset` (self)
Reset data back to original state.
- def `info` (self, key=None)
Get information about data wrapper.
- def `__len__` (self)
Get length of wrapped data.
- def `getRunID` (self)
Get the Run ID.

Public Attributes

- `data_names`
- `error_names`
- `data`
- `results`
- `constants`
- `run_id`
- `meta_data`

6.59.1 Detailed Description

Data wrapper for series data using a dictionary of data frames.

6.59.2 Member Function Documentation

6.59.2.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.59.2.2 `addResult()`

```
def skdaccess.framework.data_class.DataWrapperBase.addResult (
    self,
    rkey,
    rres ) [inherited]
```

Add a result to the data wrapper.

Parameters

<i>rkey</i>	Result key
<i>rres</i>	Result

6.59.2.3 get()

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.59.2.4 getIndices()

```
def skdaccess.framework.data_class.SeriesDictionaryWrapper.getIndices (
    self )
```

Get the indices of the data.

Returns

index of data

6.59.2.5 getIterator()

```
def skdaccess.framework.data_class.SeriesDictionaryWrapper.getIterator (
    self )
```

Get an iterator to the data.

Returns

Iterator (label, data, errors) that will cycle over data and error names

6.59.2.6 `getLength()`

```
def skdaccess.framework.data_class.SeriesDictionaryWrapper.getLength (
    self )
```

Get total number of series that the iterate will loop over.

Returns

Number of series iterator will traverse over

6.59.2.7 `getResults()`

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.59.2.8 `getRunID()`

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.59.2.9 `info()`

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.59.2.10 reset()

```
def skdaccess.framework.data_class.DataWrapperBase.reset (
    self ) [inherited]
```

Reset data back to original state.

6.59.2.11 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.59.2.12 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.59.3 Member Data Documentation

6.59.3.1 constants

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.59.3.2 data

`skdaccess.framework.data_class.DataWrapperBase.data` [inherited]

6.59.3.3 data_names

`skdaccess.framework.data_class.SeriesWrapper.data_names` [inherited]

6.59.3.4 error_names

`skdaccess.framework.data_class.SeriesWrapper.error_names` [inherited]

6.59.3.5 meta_data

`skdaccess.framework.data_class.DataWrapperBase.meta_data` [inherited]

6.59.3.6 results

`skdaccess.framework.data_class.DataWrapperBase.results` [inherited]

6.59.3.7 run_id

`skdaccess.framework.data_class.DataWrapperBase.run_id` [inherited]

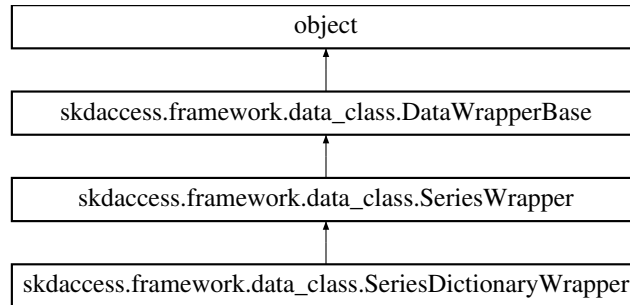
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.60 skdaccess.framework.data_class.SeriesWrapper Class Reference

Data wrapper for series data using a data panel.

Inheritance diagram for skdaccess.framework.data_class.SeriesWrapper:



Public Member Functions

- def `__init__` (self, obj_wrap, data_names, error_names=None, meta_data=None, run_id=-1)
Initialize Series Wrapper.
- def `getIterator` (self)
Get an iterator to the data.
- def `getIndices` (self)
Get the indices of the data.
- def `getLength` (self)
Get total number of series that the iterate will loop over.
- def `update` (self, obj)
Updated wrapped data.
- def `updateMetadata` (self, new_metadata)
Update metadata.
- def `get` (self)
Retrieve stored data.
- def `getResults` (self)
Retrieve accumulated results, if any.
- def `addResult` (self, rkey, rres)
Add a result to the data wrapper.
- def `reset` (self)
Reset data back to original state.
- def `info` (self, key=None)
Get information about data wrapper.
- def `__len__` (self)
Get length of wrapped data.
- def `getRunID` (self)
Get the Run ID.

Public Attributes

- [data_names](#)
- [error_names](#)
- [data](#)
- [results](#)
- [constants](#)
- [run_id](#)
- [meta_data](#)

6.60.1 Detailed Description

Data wrapper for series data using a data panel.

6.60.2 Constructor & Destructor Documentation

6.60.2.1 `__init__()`

```
def skdaccess.framework.data_class.SeriesWrapper.__init__ (
    self,
    obj_wrap,
    data_names,
    error_names = None,
    meta_data = None,
    run_id = -1 )
```

Initialize Series Wrapper.

Parameters

<i>obj_wrap</i>	Pandas data panel to wrap
<i>data_names</i>	List of data column names
<i>error_names</i>	List of error column names
<i>meta_data</i>	Metadata
<i>run_id</i>	ID of run

6.60.3 Member Function Documentation

6.60.3.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.60.3.2 `addResult()`

```
def skdaccess.framework.data_class.DataWrapperBase.addResult (
    self,
    rkey,
    rres ) [inherited]
```

Add a result to the data wrapper.

Parameters

<i>rkey</i>	Result key
<i>rres</i>	Result

6.60.3.3 `get()`

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.60.3.4 `getIndices()`

```
def skdaccess.framework.data_class.SeriesWrapper.getIndices (
    self )
```

Get the indices of the data.

Returns

index of data

6.60.3.5 `getIterator()`

```
def skdaccess.framework.data_class.SeriesWrapper.getIterator (
    self )
```

Get an iterator to the data.

Returns

Iterator (label, data, errors) that will cycle over data and error names

6.60.3.6 `getLength()`

```
def skdaccess.framework.data_class.SeriesWrapper.getLength (
    self )
```

Get total number of series that the iterate will loop over.

Returns

Number of series iterator will traverse over

6.60.3.7 `getResults()`

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.60.3.8 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.60.3.9 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.60.3.10 reset()

```
def skdaccess.framework.data_class.DataWrapperBase.reset (
    self ) [inherited]
```

Reset data back to original state.

6.60.3.11 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.60.3.12 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.60.4 Member Data Documentation**6.60.4.1 constants**

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.60.4.2 data

```
skdaccess.framework.data_class.DataWrapperBase.data [inherited]
```

6.60.4.3 data_names

```
skdaccess.framework.data_class.SeriesWrapper.data_names
```

6.60.4.4 error_names

```
skdaccess.framework.data_class.SeriesWrapper.error_names
```

6.60.4.5 meta_data

```
skdaccess.framework.data_class.DataWrapperBase.meta_data [inherited]
```

6.60.4.6 results

```
skdaccess.framework.data_class.DataWrapperBase.results [inherited]
```

6.60.4.7 run_id

```
skdaccess.framework.data_class.DataWrapperBase.run_id [inherited]
```

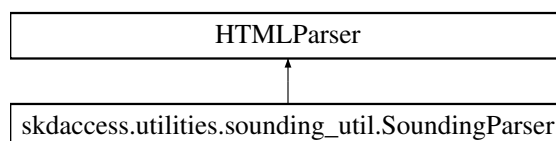
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.61 skdaccess.utilities.sounding_util.SoundingParser Class Reference

This class parses Wyoming Sounding data.

Inheritance diagram for skdaccess.utilities.sounding_util.SoundingParser:



Public Member Functions

- `def __init__ (self)`
Initialize [SoundingParser](#).
- `def handle_starttag (self, tag, attrs)`
Function called everytime a start tag is encountered.
- `def handle_endtag (self, tag)`
Function called everytime an end tag is encountered.
- `def handle_data (self, data)`
Function to parse data between <pre> tags.

Public Attributes

- `data_dict`
- `metadata_dict`
- `label`
- `in_pre_tag`
- `in_header`
- `read_data`
- `tmp`

6.61.1 Detailed Description

This class parses Wyoming Sounding data.

6.61.2 Constructor & Destructor Documentation

6.61.2.1 `__init__()`

```
def skdaccess.utilities.sounding_util.SoundingParser.__init__ (  
    self )
```

Initialize [SoundingParser](#).

6.61.3 Member Function Documentation

6.61.3.1 `handle_data()`

```
def skdaccess.utilities.sounding_util.SoundingParser.handle_data (  
    self,  
    data )
```

Function to parse data between <pre> tags.

Parameters

<i>data</i>	Input data
-------------	------------

6.61.3.2 handle_endtag()

```
def skdaccess.utilities.sounding_util.SoundingParser.handle_endtag (
    self,
    tag )
```

Function called everytime an end tag is encountered.

Parameters

<i>tag</i>	Ending tag
------------	------------

6.61.3.3 handle_starttag()

```
def skdaccess.utilities.sounding_util.SoundingParser.handle_starttag (
    self,
    tag,
    attrs )
```

Function called everytime a start tag is encountered.

Parameters

<i>tag</i>	Starting tag
<i>attrs</i>	Tag attributes

6.61.4 Member Data Documentation

6.61.4.1 data_dict

```
skdaccess.utilities.sounding_util.SoundingParser.data_dict
```

6.61.4.2 in_header

`skdaccess.utilities.sounding_util.SoundingParser.in_header`

6.61.4.3 in_pre_tag

`skdaccess.utilities.sounding_util.SoundingParser.in_pre_tag`

6.61.4.4 label

`skdaccess.utilities.sounding_util.SoundingParser.label`

6.61.4.5 metadata_dict

`skdaccess.utilities.sounding_util.SoundingParser.metadata_dict`

6.61.4.6 read_data

`skdaccess.utilities.sounding_util.SoundingParser.read_data`

6.61.4.7 tmp

`skdaccess.utilities.sounding_util.SoundingParser.tmp`

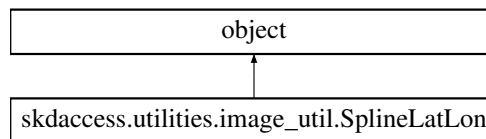
The documentation for this class was generated from the following file:

- [utilities/sounding_util.py](#)

6.62 skdaccess.utilities.image_util.SplineLatLon Class Reference

Holds a 2d spline for interpolating lat/lon grid.

Inheritance diagram for skdaccess.utilities.image_util.SplineLatLon:



Public Member Functions

- `def __init__ (self, lat_func=None, lon_func=None, lat_grid=None, lon_grid=None, x_points=None, y_points=None, lat_extents=None, lon_extents=None, y_num_pixels=None, x_num_pixels=None, x_offset=0, y_offset=0, interp_type='grid')`
Initialize [SplineLatLon](#) with premade lat/lon functions or information about the latitude and longitude.
- `def __call__ (self, y, x)`
Convert pixel coordinates to lat/lon.

Public Attributes

- [lat_func](#)
- [lon_func](#)
- [x_offset](#)
- [y_offset](#)

6.62.1 Detailed Description

Holds a 2d spline for interpolating lat/lon grid.

6.62.2 Constructor & Destructor Documentation

6.62.2.1 `__init__()`

```
def skdaccess.utilities.image_util.SplineLatLon.__init__ (
    self,
    lat_func = None,
    lon_func = None,
    lat_grid = None,
    lon_grid = None,
    x_points = None,
    y_points = None,
    lat_extents = None,
    lon_extents = None,
    y_num_pixels = None,
    x_num_pixels = None,
    x_offset = 0,
    y_offset = 0,
    interp_type = 'grid' )
```

Initialize [SplineLatLon](#) with premade lat/lon functions or information about the latitude and longitude.

Parameters

<i>lat_func</i>	Latitude spline function
<i>lon_func</i>	Longitude spline function
<i>lat_grid</i>	Latitude grid
<i>lon_grid</i>	Longitude grid
<i>x_points</i>	1d array of x coordinates
<i>y_points</i>	1d array of y coordinates
<i>lon_extents</i>	Extent of data in longitude
<i>lat_extents</i>	Extent of data in latitude
<i>y_num_pixels</i>	Number of y coordinates
<i>x_num_pixels</i>	Number of x coordinates
<i>x_offset</i>	Offset in the x coordinate
<i>y_offset</i>	Offset in the y coordinate
<i>interp_type</i>	Interpolate type. Currently only 'grid' type is supported

6.62.3 Member Function Documentation

6.62.3.1 `__call__()`

```
def skdaccess.utilities.image_util.SplineLatLon.__call__ (
    self,
    y,
    x )
```

Convert pixel coordinates to lat/lon.

Parameters

y	y coordinate
x	x coordinate

Returns

(lat, lon)

6.62.4 Member Data Documentation**6.62.4.1 lat_func**

`skdaccess.utilities.image_util.SplineLatLon.lat_func`

6.62.4.2 lon_func

`skdaccess.utilities.image_util.SplineLatLon.lon_func`

6.62.4.3 x_offset

`skdaccess.utilities.image_util.SplineLatLon.x_offset`

6.62.4.4 y_offset

`skdaccess.utilities.image_util.SplineLatLon.y_offset`

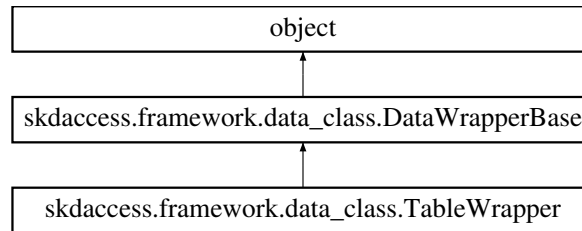
The documentation for this class was generated from the following file:

- [utilities/image_util.py](#)

6.63 skdaccess.framework.data_class.TableWrapper Class Reference

Data wrapper for table data using an ordered dictionary.

Inheritance diagram for skdaccess.framework.data_class.TableWrapper:



Public Member Functions

- def `__init__` (self, obj_wrap, run_id=-1, meta_data=None, default_columns=None, default_error_columns=None)
Construct object from input data.
- def `getIterator` (self)
Iterator access to data.
- def `getLength` (self)
Get number of data frames.
- def `updateData` (self, label, index, column_names, new_data)
Update wrapped data.
- def `addColumn` (self, label, column_names, new_data)
Add new column to data.
- def `getDefaultColumns` (self)
Get the default columns of data.
- def `getDefaultErrorColumns` (self)
Get the default error columns of data.
- def `removeFrames` (self, label_list)
Remove Data Frames from wrapper.
- def `updateFrames` (self, label_list, frame_list)
Update data frames.
- def `update` (self, obj)
Updated wrapped data.
- def `updateMetadata` (self, new_metadata)
Update metadata.
- def `get` (self)
Retrieve stored data.
- def `getResults` (self)
Retrieve accumulated results, if any.
- def `addResult` (self, rkey, rres)
Add a result to the data wrapper.
- def `reset` (self)
Reset data back to original state.

- def `info` (self, key=None)
Get information about data wrapper.
- def `__len__` (self)
Get length of wrapped data.
- def `getRunID` (self)
Get the Run ID.

Public Attributes

- `default_columns`
- `default_error_columns`
- `data`
- `results`
- `constants`
- `run_id`
- `meta_data`

6.63.1 Detailed Description

Data wrapper for table data using an ordered dictionary.

6.63.2 Constructor & Destructor Documentation

6.63.2.1 `__init__()`

```
def skdaccess.framework.data_class.TableWrapper.__init__ (
    self,
    obj_wrap,
    run_id = -1,
    meta_data = None,
    default_columns = None,
    default_error_columns = None )
```

Construct object from input data.

Parameters

<code>obj_wrap</code>	Data to be wrapped
<code>run_id</code>	ID of the run
<code>meta_data</code>	Metadata to store with data
<code>default_columns</code>	Default columns for pipeline items
<code>default_error_columns</code>	Default error columns for pipeline items

6.63.3 Member Function Documentation

6.63.3.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.63.3.2 `addColumn()`

```
def skdaccess.framework.data_class.TableWrapper.addColumn (
    self,
    label,
    column_names,
    new_data )
```

Add new column to data.

Parameters

<i>label</i>	Data label
<i>column_names</i>	Names of columns to update
<i>new_data</i>	New data to add

6.63.3.3 `addResult()`

```
def skdaccess.framework.data_class.DataWrapperBase.addResult (
    self,
    rkey,
    rres ) [inherited]
```

Add a result to the data wrapper.

Parameters

<i>rkey</i>	Result key
<i>rres</i>	Result

6.63.3.4 get()

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.63.3.5 getDefaultColumns()

```
def skdaccess.framework.data_class.TableWrapper.getDefaultColumns (
    self )
```

Get the default columns of data.

Returns

List of default columns

6.63.3.6 getDefaultErrorColumns()

```
def skdaccess.framework.data_class.TableWrapper.getDefaultErrorColumns (
    self )
```

Get the default error columns of data.

Returns

List of default error columns

6.63.3.7 `getIterator()`

```
def skdaccess.framework.data_class.TableWrapper.getIterator (
    self )
```

Iterator access to data.

Returns

iterator to (label, data frame) from Dictionary

6.63.3.8 `getLength()`

```
def skdaccess.framework.data_class.TableWrapper.getLength (
    self )
```

Get number of data frames.

Returns

Number of data frames

6.63.3.9 `getResults()`

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.63.3.10 `getRunID()`

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.63.3.11 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.63.3.12 removeFrames()

```
def skdaccess.framework.data_class.TableWrapper.removeFrames (
    self,
    label_list )
```

Remove Data Frames from wrapper.

Parameters

<i>label_list</i>	List of labels to remove
-------------------	--------------------------

6.63.3.13 reset()

```
def skdaccess.framework.data_class.DataWrapperBase.reset (
    self ) [inherited]
```

Reset data back to original state.

6.63.3.14 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.63.3.15 updateData()

```
def skdaccess.framework.data_class.TableWrapper.updateData (
    self,
    label,
    index,
    column_names,
    new_data )
```

Update wrapped data.

Parameters

<i>label</i>	Data label
<i>index</i>	Index of data to update
<i>column_names</i>	Names of columns to update
<i>new_data</i>	Data to replace the old data

6.63.3.16 updateFrames()

```
def skdaccess.framework.data_class.TableWrapper.updateFrames (
    self,
    label_list,
    frame_list )
```

Update data frames.

Parameters

<i>label_list</i>	List of labels to update
<i>frame_list</i>	List of updated frames

6.63.3.17 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
```

```
self,  
new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.63.4 Member Data Documentation

6.63.4.1 constants

`skdaccess.framework.data_class.DataWrapperBase.constants` [inherited]

6.63.4.2 data

`skdaccess.framework.data_class.DataWrapperBase.data` [inherited]

6.63.4.3 default_columns

`skdaccess.framework.data_class.TableWrapper.default_columns`

6.63.4.4 default_error_columns

`skdaccess.framework.data_class.TableWrapper.default_error_columns`

6.63.4.5 meta_data

`skdaccess.framework.data_class.DataWrapperBase.meta_data` [inherited]

6.63.4.6 results

`skdaccess.framework.data_class.DataWrapperBase.results` [inherited]

6.63.4.7 run_id

`skdaccess.framework.data_class.DataWrapperBase.run_id` [inherited]

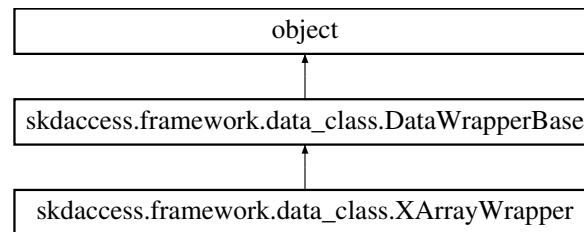
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.64 skdaccess.framework.data_class.XArrayWrapper Class Reference

Wrapper for xarrays.

Inheritance diagram for `skdaccess.framework.data_class.XArrayWrapper`:



Public Member Functions

- `def __init__(self, obj_wrap, index_list, run_id=-1)`
- `def getIterator(self)`
Get an iterator that iterates over the index.
- `def info(self, key=None)`
Get information about xarray data wrapper.
- `def update(self, obj)`
Updated wrapped data.
- `def updateMetadata(self, new_metadata)`
Update metadata.
- `def get(self)`
Retrieve stored data.
- `def getResults(self)`
Retrieve accumulated results, if any.
- `def addResult(self, rkey, rres)`
Add a result to the data wrapper.
- `def reset(self)`
Reset data back to original state.
- `def __len__(self)`
Get length of wrapped data.
- `def getRunID(self)`
Get the Run ID.

Public Attributes

- [index_list](#)
- [data](#)
- [results](#)
- [constants](#)
- [run_id](#)
- [meta_data](#)

6.64.1 Detailed Description

Wrapper for xarrays.

6.64.2 Constructor & Destructor Documentation

6.64.2.1 `__init__()`

```
def skdaccess.framework.data_class.XArrayWrapper.__init__ (
    self,
    obj_wrap,
    index_list,
    run_id = -1 )
```

6.64.3 Member Function Documentation

6.64.3.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.64.3.2 `addResult()`

```
def skdaccess.framework.data_class.DataWrapperBase.addResult (
    self,
    rkey,
    rres ) [inherited]
```

Add a result to the data wrapper.

Parameters

<i>rkey</i>	Result key
<i>rres</i>	Result

6.64.3.3 get()

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.64.3.4 getIterator()

```
def skdaccess.framework.data_class.XArrayWrapper.getIterator (
    self )
```

Get an iterator that iterates over the index.

Returns

iterator to data

6.64.3.5 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.64.3.6 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.64.3.7 info()

```
def skdaccess.framework.data_class.XArrayWrapper.info (
    self,
    key = None )
```

Get information about xarray data wrapper.

Returns

The stored metadata

6.64.3.8 reset()

```
def skdaccess.framework.data_class.DataWrapperBase.reset (
    self ) [inherited]
```

Reset data back to original state.

6.64.3.9 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.64.3.10 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.64.4 Member Data Documentation**6.64.4.1 constants**

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.64.4.2 data

```
skdaccess.framework.data_class.DataWrapperBase.data [inherited]
```

6.64.4.3 index_list

```
skdaccess.framework.data_class.XArrayWrapper.index_list
```

6.64.4.4 meta_data

`skdaccess.framework.data_class.DataWrapperBase.meta_data` [inherited]

6.64.4.5 results

`skdaccess.framework.data_class.DataWrapperBase.results` [inherited]

6.64.4.6 run_id

`skdaccess.framework.data_class.DataWrapperBase.run_id` [inherited]

The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

Chapter 7

File Documentation

7.1 astro/spectra/stream.py File Reference

Classes

- class [skdaccess.astro.spectra.stream.DataFetcher](#)
Data Fetcher for Sloan Digital Sky Survey spectra.

Namespaces

- [skdaccess.astro.spectra.stream](#)

7.2 finance/timeseries/stream.py File Reference

Classes

- class [skdaccess.finance.timeseries.stream.DataFetcher](#)
Data Fetcher for retrieving stock data.

Namespaces

- [skdaccess.finance.timeseries.stream](#)

7.3 engineering/webcam/mit_sailing/stream.py File Reference

Classes

- class [skdaccess.engineering.webcam.mit_sailing.stream.DataFetcher](#)
Data Fetcher for retrieving webcam images from the MIT Sailing Pavilion.

Namespaces

- [skdaccess.engineering.webcam.mit_sailing.stream](#)

7.4 engineering/la/traffic_counts/stream.py File Reference

Classes

- class [skdaccess.engineering.la.traffic_counts.stream.DataFetcher](#)
DataFetcher for retrieving traffic counts from LA.

Namespaces

- [skdaccess.engineering.la.traffic_counts.stream](#)

7.5 engineering/la/generic/stream.py File Reference

Classes

- class [skdaccess.engineering.la.generic.stream.DataFetcher](#)
Class for handling data requests to data.lacity.org.

Namespaces

- [skdaccess.engineering.la.generic.stream](#)

7.6 astro/tess/generic/cache.py File Reference

Classes

- class [skdaccess.astro.tess.generic.cache.DataFetcher](#)
Data Fetcher for TESS data alerts.

Namespaces

- [skdaccess.astro.tess.generic.cache](#)

7.7 astro/tess/data/cache.py File Reference

Classes

- class [skdaccess.astro.tess.data.cache.DataFetcher](#)
Data Fetcher for TESS data alerts.

Namespaces

- [skdaccess.astro.tess.data.cache](#)

7.8 astro/tess/simulated/cache.py File Reference

Classes

- class [skdaccess.astro.tess.simulated.cache.DataFetcher](#)
Data Fetcher for TESS data alerts.

Namespaces

- [skdaccess.astro.tess.simulated.cache](#)

7.9 examples/terminal_groundwater_example.py File Reference

Namespaces

- [terminal_groundwater_example](#)

Variables

- [terminal_groundwater_example.fullIDF](#)
- [terminal_groundwater_example.fullIDW](#) = fullIDF.output()
- [terminal_groundwater_example.meta_data](#) = WDF.getStationMetadata()
- [terminal_groundwater_example.dataIt](#) = fullIDW.getIterator()
- [terminal_groundwater_example.label_1](#)
- [terminal_groundwater_example.data_1](#)
- [terminal_groundwater_example.label_2](#)
- [terminal_groundwater_example.data_2](#)
- [terminal_groundwater_example.color](#)

7.10 framework/data_class.py File Reference

Classes

- class [skdaccess.framework.data_class.DataFetcherBase](#)
Base class for all data fetchers.
- class [skdaccess.framework.data_class.DataFetcherLocal](#)
Data fetcher base class for use when storing data locally.
- class [skdaccess.framework.data_class.DataFetcherStorage](#)
Data fetcher base class for use when entire data set is downloaded.
- class [skdaccess.framework.data_class.DataFetcherStream](#)
Data fetcher base class for downloading data into memory.
- class [skdaccess.framework.data_class.DataFetcherCache](#)
Data fetcher base class for downloading data and caching results on hard disk.
- class [skdaccess.framework.data_class.DataWrapperBase](#)
Base class for wrapping data for use in DiscoveryPipeline.
- class [skdaccess.framework.data_class.SeriesWrapper](#)
Data wrapper for series data using a data panel.
- class [skdaccess.framework.data_class.SeriesDictionaryWrapper](#)
Data wrapper for series data using a dictionary of data frames.
- class [skdaccess.framework.data_class.TableWrapper](#)
Data wrapper for table data using an ordered dictionary.
- class [skdaccess.framework.data_class.ImageWrapper](#)
Wrapper for image data.
- class [skdaccess.framework.data_class.XArrayWrapper](#)
Wrapper for xarrays.

Namespaces

- [skdaccess.framework.data_class](#)

7.11 framework/param_class.py File Reference

Classes

- class [skdaccess.framework.param_class.AutoParam](#)
Defines a tunable parameter class inherited by specific subclasses.
- class [skdaccess.framework.param_class.AutoParamMinMax](#)
A tunable parameter with min and max ranges, perturbs to a random value in range.
- class [skdaccess.framework.param_class.AutoParamList](#)
A tunable parameter with a specified list of choices that can be randomly selected via perturb.
- class [skdaccess.framework.param_class.AutoParamListCycle](#)
Cycles through a list of paramters.
- class [skdaccess.framework.param_class.AutoList](#)

Specifies a list for returning selections of lists, as opposed to a single element.

- class [skdaccess.framework.param_class.AutoListSubset](#)
An [AutoList](#) perturber that creates random subsets of a list.
- class [skdaccess.framework.param_class.AutoListPermute](#)
A perturber that permutes a list.
- class [skdaccess.framework.param_class.AutoListRemove](#)
Removes a different single element from the initial list at each perturb call.
- class [skdaccess.framework.param_class.AutoListCycle](#)
An Autolist that cycles through different lists.

Namespaces

- [skdaccess.framework.param_class](#)

7.12 geo/era_interim/cache/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.era_interim.cache.DataFetcher](#)
[DataFetcher](#) for retrieving ERA-I data.

Namespaces

- [skdaccess.geo.era_interim.cache.data_fetcher](#)

7.13 geo/modis/cache/reflectance/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.modis.cache.reflectance.DataFetcher](#)
Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

Namespaces

- [skdaccess.geo.modis.cache.reflectance.data_fetcher](#)

7.14 geo/modis/cache/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.modis.cache.DataFetcher](#)
Data Fetcher for MODIS data.

Namespaces

- [skdaccess.geo.modis.cache.data_fetcher](#)

7.15 geo/modis/cache/cloud_opacity/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.modis.cache.cloud_opacity.DataFetcher](#)
Data Fetcher for MODIS Cloud Opacity.

Namespaces

- [skdaccess.geo.modis.cache.cloud_opacity.data_fetcher](#)

7.16 geo/modis/cache/cloud_mask/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.modis.cache.cloud_mask.DataFetcher](#)
Data Fetcher for MODIS Cloud Mask.

Namespaces

- [skdaccess.geo.modis.cache.cloud_mask.data_fetcher](#)

7.17 geo/modis/stream/reflectance/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.modis.stream.reflectance.DataFetcher](#)
Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

Namespaces

- [skdaccess.geo.modis.stream.reflectance.data_fetcher](#)

7.18 geo/modis/stream/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.modis.stream.DataFetcher](#)
Data Fetcher for MODIS data.

Namespaces

- [skdaccess.geo.modis.stream.data_fetcher](#)

7.19 geo/modis/stream/cloud_opacity/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.modis.stream.cloud_opacity.DataFetcher](#)
Data Fetcher for MODIS Cloud Opacity.

Namespaces

- [skdaccess.geo.modis.stream.cloud_opacity.data_fetcher](#)

7.20 geo/modis/stream/cloud_mask/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.modis.stream.cloud_mask.DataFetcher](#)
Data Fetcher for MODIS Cloud Mask.

Namespaces

- [skdaccess.geo.modis.stream.cloud_mask.data_fetcher](#)

7.21 geo/gldas/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.gldas.DataFetcher](#)
Data Fetcher for GLDAS data.

Namespaces

- [skdaccess.geo.gldas.data_fetcher](#)

7.22 geo/uavsar/cache/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.uavsar.cache.DataFetcher](#)
Data Fetcher for UAVSAR data.

Namespaces

- [skdaccess.geo.uavsar.cache.data_fetcher](#)

7.23 geo/pbo/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.pbo.DataFetcher](#)
Data fetcher for PBO GPS data.

Namespaces

- [skdaccess.geo.pbo.data_fetcher](#)

7.24 geo/grace/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.grace.DataFetcher](#)
Data Fetcher for GRACE data.

Namespaces

- [skdaccess.geo.grace.data_fetcher](#)

7.25 geo/grace/mascon/cache/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.grace.mascon.cache.DataFetcher](#)
Data Fetcher for GRACE mascon data.

Namespaces

- [skdaccess.geo.grace.mascon.cache.data_fetcher](#)

7.26 geo/sentinel_1/cache/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.sentinel_1.cache.DataFetcher](#)
[DataFetcher](#) for retrieving Sentinel SLC data.

Namespaces

- [skdaccess.geo.sentinel_1.cache.data_fetcher](#)

7.27 geo/groundwater/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.groundwater.DataFetcher](#)
Generates Data Wrappers of groundwater measurements taken in the US.

Namespaces

- [skdaccess.geo.groundwater.data_fetcher](#)

7.28 geo/srtm/cache/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.srtm.cache.DataFetcher](#)
[DataFetcher](#) for retrieving data from the Shuttle Radar Topography Mission.

Namespaces

- [skdaccess.geo.srtm.cache.data_fetcher](#)

7.29 geo/mahali/temperature/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.mahali.temperature.DataFetcher](#)
Data Fetcher for Mahali temperature data.

Namespaces

- [skdaccess.geo.mahali.temperature.data_fetcher](#)

7.30 geo/mahali/rinex/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.mahali.rinex.DataFetcher](#)
Data Fetcher for Mahali Data.

Namespaces

- [skdaccess.geo.mahali.rinex.data_fetcher](#)

7.31 geo/mahali/tec/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.mahali.tec.DataFetcher](#)
Data Fetcher for Mahali Data.

Namespaces

- [skdaccess.geo.mahali.tec.data_fetcher](#)

7.32 geo/magnetometer/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.magnetometer.DataFetcher](#)
Data fetcher for USGS geomagnetic observatories.

Namespaces

- [skdaccess.geo.magnetometer.data_fetcher](#)

7.33 geo/ngl_gps/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.ngl_gps.DataFetcher](#)
Data fetcher for GPS data from Nevada Geodetic Laboratory.

Namespaces

- [skdaccess.geo.ngl_gps.data_fetcher](#)

7.34 geo/wyoming_sounding/cache/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.wyoming_sounding.cache.DataFetcher](#)
[DataFetcher](#) for retrieving Wyoming Sounding data.

Namespaces

- [skdaccess.geo.wyoming_sounding.cache.data_fetcher](#)

7.35 geo/wyoming_sounding/stream/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.wyoming_sounding.stream.DataFetcher](#)
[DataFetcher](#) for retrieving Wyoming Sounding data.

Namespaces

- [skdaccess.geo.wyoming_sounding.stream.data_fetcher](#)

7.36 geo/imsdnhs/data_fetcher.py File Reference

Classes

- class [skdaccess.geo.imsdnhs.DataFetcher](#)
Fetches data for the Interactive Multisensor Snow and Ice Mapping System Daily Northern Hemisphere Snow and Ice Analysis.

Namespaces

- [skdaccess.geo.imsdnhs.data_fetcher](#)

7.37 astro/kepler/data_fetcher.py File Reference

Classes

- class [skdaccess.astro.kepler.DataFetcher](#)
Data Fetcher for Kepler light curve data.

Namespaces

- [skdaccess.astro.kepler.data_fetcher](#)

7.38 astro/voyager/data_fetcher.py File Reference

Classes

- class [skdaccess.astro.voyager.DataFetcher](#)
Data Fetcher for Mahali temperature data.

Namespaces

- [skdaccess.astro.voyager.data_fetcher](#)

7.39 solar/sdo/data_fetcher.py File Reference

Classes

- class [skdaccess.solar.sdo.DataFetcher](#)
Data Fetcher for the Solar Dynamics Observatory.

Namespaces

- [skdaccess.solar.sdo.data_fetcher](#)

7.40 planetary/ode/cache/data_fetcher.py File Reference

Classes

- class [skdaccess.planetary.ode.cache.DataFetcher](#)
Data Fetcher from the Orbital Data Explorer (ODE)

Namespaces

- [skdaccess.planetary.ode.cache.data_fetcher](#)

7.41 geo/mahali/rinex/data_wrapper.py File Reference

Classes

- class [skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper](#)
Data wrapper for Mahali data.

Namespaces

- [skdaccess.geo.mahali.rinex.data_wrapper](#)

7.42 utilities/file_browser.py File Reference

Classes

- class [skdaccess.utilities.file_browser.FileBrowser](#)

Namespaces

- [skdaccess.utilities.file_browser](#)

7.43 utilities/file_util.py File Reference

Namespaces

- [skdaccess.utilities.file_util](#)

Functions

- def [skdaccess.utilities.file_util.openPandasHDFStoreLocking](#) (filename, mode)
Open a pandas HDF store that may be locked:

7.44 utilities/grace_util.py File Reference

Namespaces

- [skdaccess.utilities.grace_util](#)

Functions

- def [skdaccess.utilities.grace_util.averageDates](#) (dates, round_nearest_day=False)
Compute the average of a pandas series of timestamps.
- def [skdaccess.utilities.grace_util.dateMismatch](#) (dates, days=10)
Check if dates are not within a certain number of days of each other.
- def [skdaccess.utilities.grace_util.computeEWD](#) (grace_data, scale_factor, round_nearest_day=False)
Compute scale corrected equivalent water depth.
- def [skdaccess.utilities.grace_util.readTellusData](#) (filename, lat_lon_list, lat_name, lon_name, data_name, data_label=None, time_name=None, lat_bounds_name=None, lon_bounds_name=None, uncertainty_name=None, lat_bounds=None, lon_bounds=None)
This function reads in netcdf data provided by GRACE Tellus.
- def [skdaccess.utilities.grace_util.getStartEndDate](#) (in_data)

7.45 utilities/gw_util.py File Reference

Namespaces

- [skdaccess.utilities.gw_util](#)

Functions

- def [skdaccess.utilities.gw_util.combine_water_heights](#) (in_data)
Combine median and average water heights.

7.46 utilities/image_util.py File Reference

Classes

- class [skdaccess.utilities.image_util.SplineLatLon](#)
Holds a 2d spline for interpolating lat/lon grid.
- class [skdaccess.utilities.image_util.LinearGeolocation](#)
This class provides functions to convert between pixel and geodetic coordinates.
- class [skdaccess.utilities.image_util.AffineGlobalCoords](#)
Convert between projected and pixel coordinates using an affine transformation.

Namespaces

- [skdaccess.utilities.image_util](#)

Functions

- def [skdaccess.utilities.image_util.SplineGeolocation](#) (object)
This class holds splines to convert between 2d cartesian and geodetic coordinates.
- def [skdaccess.utilities.image_util.getExtentsFromCentersPlateCarree](#) (westmost_pixel_lon, eastmost_pixel_lon, southmost_pixel_lat, northmost_pixel_lat, lon_grid_spacing, lat_grid_spacing)
- def [skdaccess.utilities.image_util.convertBinCentersToEdges](#) (bin_centers, dtype=None)
Calculate edges of a set of bins from their centers.
- def [skdaccess.utilities.image_util.getGeoTransform](#) (extents, x_size, y_size, y_flipped=True)
Get 6 geotransform coefficients from the extents of an image and its shape.

Variables

- [skdaccess.utilities.image_util.x_offset](#)
- [skdaccess.utilities.image_util.y_offset](#)
- [skdaccess.utilities.image_util.lat_spline](#)
- [skdaccess.utilities.image_util.lon_spline](#)
- [skdaccess.utilities.image_util.x_spline](#)
- [skdaccess.utilities.image_util.y_spline](#)

7.47 utilities/kepler_util.py File Reference

Namespaces

- [skdaccess.utilities.kepler_util](#)

Functions

- def [skdaccess.utilities.kepler_util.normalize](#) (in_data, column='PDCSAP_FLUX', group_column='QUARTER')
This function normalizes PDCSAP_FLUX data by quarter by dividing the flux of each quarter by the median of that respective quarter.

7.48 utilities/mahali_util.py File Reference

Namespaces

- [skdaccess.utilities.mahali_util](#)

Functions

- def [skdaccess.utilities.mahali_util.convert_date](#) (in_date)
Converts input string to pandas date time, ignores other types of objects.
- def [skdaccess.utilities.mahali_util.parseIonoFile](#) (in_file, compression='infer')

7.49 utilities/modis_util.py File Reference

Classes

- class [skdaccess.utilities.modis_util.LatLon](#)
Calculates Lat/Lon position from y,x pixel coordinate.

Namespaces

- [skdaccess.utilities.modis_util](#)

Functions

- def [skdaccess.utilities.modis_util.getImageType](#) (in_data)
Determine what type of modis data is being processed.
- def [skdaccess.utilities.modis_util.calibrateModis](#) (data, metadata)
This function calibrates input modis data.
- def [skdaccess.utilities.modis_util.rescale](#) (in_array, max_val=0.9, min_val=-0.01)
This function rescales an image to fall between 0 and 1.
- def [skdaccess.utilities.modis_util.checkBit](#) (data, bit)
Get the bit value from a bit flag.
- def [skdaccess.utilities.modis_util.createGrid](#) (data, y_start, y_end, x_start, x_end, y_grid, x_grid, dtype, grid_fill=np.nan)
Subsets image data into a smaller image.
- def [skdaccess.utilities.modis_util.getFileIDs](#) (modis_identifier, start_date, end_date, lat, lon, daynightboth)
Retrieve file IDs for images matching search parameters.
- def [skdaccess.utilities.modis_util.getFileURLs](#) (file_ids)
Retrieve the ftp location for a list of file IDs.
- def [skdaccess.utilities.modis_util.getModisData](#) (dataset, variable_name)
Loads modis data.
- def [skdaccess.utilities.modis_util.readMODISData](#) (modis_list, variables, grid, grid_fill, use_long_name, platform, product_id)
Retrieve a list of modis data.

7.50 utilities/ode_util.py File Reference

Namespaces

- [skdaccess.utilities.ode_util](#)

Functions

- def [skdaccess.utilities.ode_util.query_yes_no](#) (question, default="yes")
- def [skdaccess.utilities.ode_util.get_query_url](#) (target, mission, instrument, product_type, western_lon, eastern_lon, min_lat, max_lat, min_ob_time, max_ob_time, product_id, query_type, output, results, number_product_limit, result_offset_number)
- def [skdaccess.utilities.ode_util.get_files_urls](#) (query_url, file_name='*', print_info=False)
- def [skdaccess.utilities.ode_util.query_files_urls](#) (target, mission, instrument, product_type, western_lon, eastern_lon, min_lat, max_lat, min_ob_time, max_ob_time, product_id, file_name, number_product_limit, result_offset_number)
Retrieve the URL locations based on a query using ODE REST interface.
- def [skdaccess.utilities.ode_util.correct_CRISM_label](#) (label_file_location)
- def [skdaccess.utilities.ode_util.correct_file_name_case_in_label](#) (label_file_location, other_file_locations)
- def [skdaccess.utilities.ode_util.correct_label_file](#) (label_file_location, other_file_locations=[])
Correct a label file if GDAL cannot open the corresponding data file.
- def [skdaccess.utilities.ode_util.get_raster_array](#) (gdal_raster, remove_ndv=True)
Get a NumPy array from a raster opened with GDAL.
- def [skdaccess.utilities.ode_util.get_raster_extent](#) (gdal_raster)
Get the extent of a raster opened with GDAL.

7.51 utilities/pbo_util.py File Reference

Namespaces

- [skdaccess.utilities.pbo_util](#)

Functions

- def [skdaccess.utilities.pbo_util.getStationCoords](#) (pbo_info, station_list)
Get the station coordinates for a list of stations.
- def [skdaccess.utilities.pbo_util.getLatLonRange](#) (pbo_info, station_list)
Retrieve the range of latitude and longitude occupied by a set of stations.
- def [skdaccess.utilities.pbo_util.getROIstations](#) (geo_point, radiusParam, data, header)
This function returns the 4ID station codes for the stations in a region.
- def [skdaccess.utilities.pbo_util.stab_sys](#) (data_iterator, metadata, stab_min_NE=.0005, stab_min_U=.005, sigsc=2, errProp=1)
Stabilize GPS data to a region.
- def [skdaccess.utilities.pbo_util.propagateErrors](#) (R, sc, stationCovs)
Propagate GPS errors.
- def [skdaccess.utilities.pbo_util.nostab_sys](#) (allH, allID, timerng, indx=1, mdyratio=.7, use_progress_bar=True, index_date_only=False)
Do not apply stabilization and simply returns stations after checking for sufficient amount of data.
- def [skdaccess.utilities.pbo_util.removeAntennaOffset](#) (antenna_offsets, data, window_start=pd.to_timedelta('4D'), window_end=pd.to_timedelta('4D'), min_diff=0.005, debug=False)
Remove offsets caused by changes in antennas.

7.52 utilities/sentinel_1_util.py File Reference

Namespaces

- [skdaccess.utilities.sentinel_1_util](#)

Functions

- def [skdaccess.utilities.sentinel_1_util.parseSatelliteData](#) (in_satellite_file)
Parse Sentinel satellite data.

7.53 utilities/sounding_util.py File Reference

Classes

- class [skdaccess.utilities.sounding_util.SoundingParser](#)
This class parses Wyoming Sounding data.

Namespaces

- [skdaccess.utilities.sounding_util](#)

Functions

- def [skdaccess.utilities.sounding_util.generateQueries](#) (station_number, year_list, month_list, day_start, day_end, start_hour, end_hour)
Generate url queries for sounding data.

7.54 utilities/srtm_util.py File Reference

Namespaces

- [skdaccess.utilities.srtm_util](#)

Functions

- def [skdaccess.utilities.srtm_util.merge_srtm_tiles](#) (srtm_tiles, lon_min, lon_max, lat_min, lat_max)
- def [skdaccess.utilities.srtm_util.getSRTMLatLon](#) (lat_min, lat_max, lon_min, lon_max)
Retrieve parameters that encompass area when creating SRTM data fetcher.
- def [skdaccess.utilities.srtm_util.getSRTMData](#) (srtmdw, lat_start, lat_end, lon_start, lon_end)
Select SRTM data in a latitude/longitude box.

7.55 utilities/support.py File Reference

Namespaces

- [skdaccess.utilities.support](#)

Functions

- def [skdaccess.utilities.support.retrieveCommonDatesHDF](#) (support_data_filename, key_list, in_date_list)
Get a list of all dates that have data available.
- def [skdaccess.utilities.support.progress_bar](#) (in_iterable, total=None, enabled=True)
Progress bar using tqdm.
- def [skdaccess.utilities.support.convertToStr](#) (in_value, zfill=0)
- def [skdaccess.utilities.support.join_string](#) (part1, part2, concatenation_string='AND', seperator=' ')
Join two strings together using a concatenation string.

7.56 utilities/tess_utils.py File Reference

Namespaces

- [skdaccess.utilities.tess_utils](#)

Functions

- def [skdaccess.utilities.tess_utils.parseTessData](#) (fits_data)
Retrieve Tess lightcurve data from astropy.io.fits.HDUList object.

7.57 utilities/uavsar_util.py File Reference

Namespaces

- [skdaccess.utilities.uavsar_util](#)

Functions

- def [skdaccess.utilities.uavsar_util.readUAVSARMetadata](#) (in_file)
Parse UAVSAR metadata.

Index

`__call__`

- `skdaccess::framework::param_class::AutoList`, 66
- `skdaccess::framework::param_class::AutoListCycle`, 71
- `skdaccess::framework::param_class::AutoList`↔
`Permute`, 75
- `skdaccess::framework::param_class::AutoList`↔
`Remove`, 79
- `skdaccess::framework::param_class::AutoList`↔
`Subset`, 83
- `skdaccess::framework::param_class::AutoParam`, 88
- `skdaccess::framework::param_class::AutoParamList`, 90
- `skdaccess::framework::param_class::AutoParam`↔
`ListCycle`, 93
- `skdaccess::framework::param_class::AutoParam`↔
`MinMax`, 96
- `skdaccess::utilities::image_util::SplineLatLon`, 408
- `skdaccess::utilities::modis_util::LatLon`, 385

`__getitem__`

- `skdaccess::framework::param_class::AutoList`, 66
- `skdaccess::framework::param_class::AutoListCycle`, 71
- `skdaccess::framework::param_class::AutoList`↔
`Permute`, 75
- `skdaccess::framework::param_class::AutoList`↔
`Remove`, 79
- `skdaccess::framework::param_class::AutoList`↔
`Subset`, 83

`__init__`

- `skdaccess::astro::kepler::data_fetcher::DataFetcher`, 134
- `skdaccess::astro::spectra::stream::DataFetcher`, 307
- `skdaccess::astro::tess::data::cache::DataFetcher`, 297
- `skdaccess::astro::tess::generic::cache::DataFetcher`, 117
- `skdaccess::astro::tess::simulated::cache::Data`↔
`Fetcher`, 113
- `skdaccess::astro::voyager::data_fetcher::Data`↔
`Fetcher`, 288
- `skdaccess::engineering::la::generic::stream::Data`↔
`Fetcher`, 143
- `skdaccess::engineering::la::traffic_counts::stream`↔
`::DataFetcher`, 199

- `skdaccess::engineering::webcam::mit_sailing`↔
`::stream::DataFetcher`, 270
- `skdaccess::finance::timeseries::stream::Data`↔
`Fetcher`, 176
- `skdaccess::framework::data_class::DataFetcher`↔
`Base`, 338
- `skdaccess::framework::data_class::DataWrapper`↔
`Base`, 372
- `skdaccess::framework::data_class::SeriesWrapper`, 398
- `skdaccess::framework::data_class::TableWrapper`, 411
- `skdaccess::framework::data_class::XArrayWrapper`, 419
- `skdaccess::framework::param_class::AutoList`, 66
- `skdaccess::framework::param_class::AutoListCycle`, 70
- `skdaccess::framework::param_class::AutoList`↔
`Remove`, 79
- `skdaccess::framework::param_class::AutoParam`, 87
- `skdaccess::framework::param_class::AutoParamList`, 90
- `skdaccess::framework::param_class::AutoParam`↔
`ListCycle`, 93
- `skdaccess::framework::param_class::AutoParam`↔
`MinMax`, 95
- `skdaccess::geo::era_interim::cache::data_fetcher::`↔
`DataFetcher`, 105
- `skdaccess::geo::gldas::data_fetcher::DataFetcher`, 193
- `skdaccess::geo::grace::data_fetcher::DataFetcher`, 264
- `skdaccess::geo::grace::mascon::cache::data`↔
`fetcher::DataFetcher`, 320
- `skdaccess::geo::groundwater::data_fetcher::Data`↔
`Fetcher`, 126
- `skdaccess::geo::imsdnhs::data_fetcher::Data`↔
`Fetcher`, 300
- `skdaccess::geo::magnetometer::data_fetcher::`↔
`DataFetcher`, 209
- `skdaccess::geo::mahali::rinex::data_fetcher::Data`↔
`Fetcher`, 184
- `skdaccess::geo::mahali::tec::data_fetcher::Data`↔
`Fetcher`, 201
- `skdaccess::geo::mahali::temperature::data_fetcher`↔

- `::DataFetcher`, 168
- `skdaccess::geo::modis::cache::cloud_mask::data_↵
fetcher::DataFetcher`, 182
- `skdaccess::geo::modis::cache::cloud_opacity↵
::data_fetcher::DataFetcher`, 173
- `skdaccess::geo::modis::cache::data_fetcher::Data↵
Fetcher`, 233
- `skdaccess::geo::modis::cache::reflectance::data_↵
fetcher::DataFetcher`, 166
- `skdaccess::geo::modis::stream::cloud_mask::data↵
_fetcher::DataFetcher`, 174
- `skdaccess::geo::modis::stream::cloud_opacity↵
::data_fetcher::DataFetcher`, 115
- `skdaccess::geo::modis::stream::data_fetcher::↵
DataFetcher`, 313
- `skdaccess::geo::modis::stream::reflectance::data_↵
fetcher::DataFetcher`, 242
- `skdaccess::geo::ngl_gps::data_fetcher::DataFetcher`, 224
- `skdaccess::geo::pbo::data_fetcher::DataFetcher`, 253
- `skdaccess::geo::sentinel_1::cache::data_fetcher::↵
DataFetcher`, 329
- `skdaccess::geo::srtm::cache::data_fetcher::Data↵
Fetcher`, 157
- `skdaccess::geo::uavsar::cache::data_fetcher::Data↵
Fetcher`, 216
- `skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher`, 244
- `skdaccess::geo::wyoming_sounding::stream::data↵
_fetcher::DataFetcher`, 150
- `skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher`, 277
- `skdaccess::solar::sdo::data_fetcher::DataFetcher`, 99
- `skdaccess::utilities::file_browser::FileBrowser`, 377
- `skdaccess::utilities::image_util::AffineGlobalCoords`, 63
- `skdaccess::utilities::image_util::LinearGeolocation`, 388
- `skdaccess::utilities::image_util::SplineLatLon`, 407
- `skdaccess::utilities::modis_util::LatLon`, 385
- `skdaccess::utilities::sounding_util::SoundingParser`, 404
- `__len__`
 - `skdaccess::framework::data_class::DataWrapper↵
Base`, 373
 - `skdaccess::framework::data_class::ImageWrapper`, 379
 - `skdaccess::framework::data_class::SeriesDictionary↵
Wrapper`, 392
 - `skdaccess::framework::data_class::SeriesWrapper`, 398
 - `skdaccess::framework::data_class::TableWrapper`, 412
 - `skdaccess::framework::data_class::XArrayWrapper`, 419
- `skdaccess::framework::param_class::AutoList`, 67
- `skdaccess::framework::param_class::AutoListCycle`, 71
- `skdaccess::framework::param_class::AutoList↵
Permute`, 75
- `skdaccess::framework::param_class::AutoList↵
Remove`, 80
- `skdaccess::framework::param_class::AutoList↵
Subset`, 84
- `skdaccess::geo::mahali::rinex::data_wrapper::Data↵
Wrapper`, 367
- `__setitem__`
 - `skdaccess::framework::param_class::AutoList`, 67
 - `skdaccess::framework::param_class::AutoListCycle`, 72
 - `skdaccess::framework::param_class::AutoList↵
Permute`, 76
 - `skdaccess::framework::param_class::AutoList↵
Remove`, 80
 - `skdaccess::framework::param_class::AutoList↵
Subset`, 84
- `__str__`
 - `skdaccess::astro::kepler::data_fetcher::DataFetcher`, 135
 - `skdaccess::astro::spectra::stream::DataFetcher`, 307
 - `skdaccess::astro::tess::generic::cache::DataFetcher`, 118
 - `skdaccess::astro::voyager::data_fetcher::Data↵
Fetcher`, 288
 - `skdaccess::engineering::la::generic::stream::Data↵
Fetcher`, 143
 - `skdaccess::engineering::webcam::mit_sailing↵
::stream::DataFetcher`, 271
 - `skdaccess::finance::timeseries::stream::Data↵
Fetcher`, 177
 - `skdaccess::framework::data_class::DataFetcher↵
Base`, 338
 - `skdaccess::framework::data_class::DataFetcher↵
Cache`, 344
 - `skdaccess::framework::data_class::DataFetcher↵
Local`, 350
 - `skdaccess::framework::data_class::DataFetcher↵
Storage`, 356
 - `skdaccess::framework::data_class::DataFetcher↵
Stream`, 362
 - `skdaccess::framework::param_class::AutoList`, 67
 - `skdaccess::framework::param_class::AutoListCycle`, 72
 - `skdaccess::framework::param_class::AutoList↵
Permute`, 76
 - `skdaccess::framework::param_class::AutoList↵
Remove`, 80
 - `skdaccess::framework::param_class::AutoList↵`

- Subset, [84](#)
- skdaccess::framework::param_class::AutoParam, [88](#)
- skdaccess::framework::param_class::AutoParamList, [90](#)
- skdaccess::framework::param_class::AutoParam↔ListCycle, [93](#)
- skdaccess::framework::param_class::AutoParam↔MinMax, [96](#)
- skdaccess::geo::era_interim::cache::data_fetcher::↔DataFetcher, [106](#)
- skdaccess::geo::gldas::data_fetcher::DataFetcher, [194](#)
- skdaccess::geo::grace::data_fetcher::DataFetcher, [264](#)
- skdaccess::geo::grace::mascon::cache::data_↔fetcher::DataFetcher, [321](#)
- skdaccess::geo::groundwater::data_fetcher::Data↔Fetcher, [127](#)
- skdaccess::geo::imsdnhs::data_fetcher::Data↔Fetcher, [301](#)
- skdaccess::geo::magnetometer::data_fetcher::↔DataFetcher, [209](#)
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔Fetcher, [185](#)
- skdaccess::geo::mahali::tec::data_fetcher::Data↔Fetcher, [202](#)
- skdaccess::geo::mahali::temperature::data_fetcher↔::DataFetcher, [168](#)
- skdaccess::geo::modis::cache::data_fetcher::Data↔Fetcher, [234](#)
- skdaccess::geo::modis::stream::data_fetcher::↔DataFetcher, [313](#)
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, [225](#)
- skdaccess::geo::pbo::data_fetcher::DataFetcher, [255](#)
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔DataFetcher, [330](#)
- skdaccess::geo::srtm::cache::data_fetcher::Data↔Fetcher, [158](#)
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔Fetcher, [217](#)
- skdaccess::geo::wyoming_sounding::cache::data_↔fetcher::DataFetcher, [245](#)
- skdaccess::geo::wyoming_sounding::stream::data↔_fetcher::DataFetcher, [150](#)
- skdaccess::planetary::ode::cache::data_fetcher::↔DataFetcher, [277](#)
- skdaccess::solar::sdo::data_fetcher::DataFetcher, [100](#)
- skdaccess::framework::data_class::DataWrapper↔Base, [373](#)
- skdaccess::framework::data_class::ImageWrapper, [380](#)
- skdaccess::framework::data_class::SeriesDictionary↔Wrapper, [392](#)
- skdaccess::framework::data_class::SeriesWrapper, [399](#)
- skdaccess::framework::data_class::TableWrapper, [412](#)
- skdaccess::framework::data_class::XArrayWrapper, [419](#)
- skdaccess::geo::mahali::rinex::data_wrapper::Data↔Wrapper, [368](#)
- alat
 - skdaccess::utilities::modis_util::LatLon, [386](#)
- alon
 - skdaccess::utilities::modis_util::LatLon, [386](#)
- antenna_info
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, [261](#)
- ap_paramList
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, [141](#)
 - skdaccess::astro::spectra::stream::DataFetcher, [311](#)
 - skdaccess::astro::tess::generic::cache::DataFetcher, [124](#)
 - skdaccess::astro::voyager::data_fetcher::Data↔Fetcher, [296](#)
 - skdaccess::engineering::la::generic::stream::Data↔Fetcher, [147](#)
 - skdaccess::engineering::webcam::mit_sailing↔::stream::DataFetcher, [274](#)
 - skdaccess::finance::timeseries::stream::Data↔Fetcher, [180](#)
 - skdaccess::framework::data_class::DataFetcher↔Base, [341](#)
 - skdaccess::framework::data_class::DataFetcher↔Cache, [349](#)
 - skdaccess::framework::data_class::DataFetcher↔Local, [354](#)
 - skdaccess::framework::data_class::DataFetcher↔Storage, [361](#)
 - skdaccess::framework::data_class::DataFetcher↔Stream, [366](#)
 - skdaccess::geo::era_interim::cache::data_fetcher::↔DataFetcher, [111](#)
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, [198](#)
 - skdaccess::geo::grace::data_fetcher::DataFetcher, [269](#)
 - skdaccess::geo::grace::mascon::cache::data_↔fetcher::DataFetcher, [326](#)
 - skdaccess::geo::groundwater::data_fetcher::Data↔Fetcher, [132](#)

addColumn

- skdaccess::framework::data_class::TableWrapper, [412](#)

addResult

- skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, [305](#)
- skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, [213](#)
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, [191](#)
- skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, [207](#)
- skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, [172](#)
- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [240](#)
- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, [317](#)
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
[230](#)
- skdaccess::geo::pbo::data_fetcher::DataFetcher, [261](#)
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [335](#)
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [163](#)
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, [222](#)
- skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, [250](#)
- skdaccess::geo::wyoming_sounding::stream::data_↔
_fetcher::DataFetcher, [154](#)
- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [284](#)
- skdaccess::solar::sdo::data_fetcher::DataFetcher,
[103](#)
- app_token
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, [147](#)
- arcsecond_sampling
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [163](#)
- astro/kepler/data_fetcher.py, [436](#)
- astro/spectra/stream.py, [425](#)
- astro/tess/data/cache.py, [427](#)
- astro/tess/generic/cache.py, [426](#)
- astro/tess/simulated/cache.py, [427](#)
- astro/voyager/data_fetcher.py, [436](#)
- averageDates
 - skdaccess::utilities::grace_util, [32](#)
- base_url
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [296](#)
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, [147](#)
- base_url_and_endpoint
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, [147](#)
- cacheData
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
[135](#)
 - skdaccess::astro::tess::generic::cache::DataFetcher,
[118](#)
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [288](#)
 - skdaccess::framework::data_class::DataFetcher↔
Cache, [344](#)
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, [106](#)
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, [321](#)
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, [185](#)
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, [202](#)
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [234](#)
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [330](#)
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [158](#)
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, [217](#)
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, [245](#)
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [278](#)
- calibrateModis
 - skdaccess::utilities::modis_util, [41](#)
- camera_list
 - skdaccess::engineering::webcam::mit_sailing↔
::stream::DataFetcher, [275](#)
- channels
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, [214](#)
- checkBit
 - skdaccess::utilities::modis_util, [41](#)
- checkIfDataExists
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
[136](#)
 - skdaccess::astro::tess::generic::cache::DataFetcher,
[119](#)
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [289](#)
 - skdaccess::framework::data_class::DataFetcher↔
Cache, [344](#)
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, [107](#)
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, [322](#)
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, [186](#)

- skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, [202](#)
- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [235](#)
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [331](#)
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [159](#)
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, [217](#)
- skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, [246](#)
- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [278](#)
- color
 - terminal_groundwater_example, [60](#)
- combine_water_heights
 - skdaccess::utilities::gw_util, [34](#)
- computeEWD
 - skdaccess::utilities::grace_util, [32](#)
- constants
 - skdaccess::framework::data_class::DataWrapper↔
Base, [376](#)
 - skdaccess::framework::data_class::ImageWrapper,
[383](#)
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, [395](#)
 - skdaccess::framework::data_class::SeriesWrapper,
[402](#)
 - skdaccess::framework::data_class::TableWrapper,
[417](#)
 - skdaccess::framework::data_class::XArrayWrapper,
[422](#)
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, [370](#)
- convert_date
 - skdaccess::utilities::mahali_util, [40](#)
- convertBinCentersToEdges
 - skdaccess::utilities::image_util, [36](#)
- convertToStr
 - skdaccess::utilities::support, [57](#)
- coordinate_dict
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, [305](#)
- correct_CRISM_label
 - skdaccess::utilities::ode_util, [46](#)
- correct_file_name_case_in_label
 - skdaccess::utilities::ode_util, [46](#)
- correct_label_file
 - skdaccess::utilities::ode_util, [46](#)
- createGrid
 - skdaccess::utilities::modis_util, [42](#)
- current_index
 - skdaccess::framework::param_class::AutoParam↔
ListCycle, [94](#)
- cutoff
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, [132](#)
- data
 - skdaccess::framework::data_class::DataWrapper↔
Base, [376](#)
 - skdaccess::framework::data_class::ImageWrapper,
[383](#)
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, [395](#)
 - skdaccess::framework::data_class::SeriesWrapper,
[402](#)
 - skdaccess::framework::data_class::TableWrapper,
[417](#)
 - skdaccess::framework::data_class::XArrayWrapper,
[422](#)
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, [370](#)
- data_1
 - terminal_groundwater_example, [60](#)
- data_2
 - terminal_groundwater_example, [60](#)
- data_dict
 - skdaccess::utilities::sounding_util::SoundingParser,
[405](#)
- data_names
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, [396](#)
 - skdaccess::framework::data_class::SeriesWrapper,
[402](#)
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, [112](#)
- data_type
 - skdaccess::finance::timeseries::stream::Data↔
Fetcher, [181](#)
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, [214](#)
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
[230](#)
- datalt
 - terminal_groundwater_example, [60](#)
- date_list
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, [112](#)
- date_range
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, [191](#)
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, [207](#)
- dateMismatch
 - skdaccess::utilities::grace_util, [33](#)
- day_end

- skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 250
- skdaccess::geo::wyoming_sounding::stream::data_↵
_fetcher::DataFetcher, 154
- day_start
 - skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 250
 - skdaccess::geo::wyoming_sounding::stream::data_↵
_fetcher::DataFetcher, 154
- daynightboth
 - skdaccess::geo::modis::cache::data_fetcher::Data_↵
Fetcher, 240
 - skdaccess::geo::modis::stream::data_fetcher::↵
DataFetcher, 317
- decimals
 - skdaccess::framework::param_class::AutoParam_↵
MinMax, 97
- default_columns
 - skdaccess::framework::data_class::TableWrapper,
417
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 261
- default_error_columns
 - skdaccess::framework::data_class::TableWrapper,
417
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 261
- deleteData
 - skdaccess::framework::data_class::ImageWrapper,
380
- dirs
 - skdaccess::utilities::file_browser::FileBrowser, 378
- downloadFullDataset
 - skdaccess::framework::data_class::DataFetcher_↵
Storage, 356
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
194
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
264
 - skdaccess::geo::groundwater::data_fetcher::Data_↵
Fetcher, 127
 - skdaccess::geo::imsdnhs::data_fetcher::Data_↵
Fetcher, 301
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
225
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 255
- downloadKeplerData
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
136
- eastern_lon
 - skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 284
- end_date
 - skdaccess::finance::timeseries::stream::Data_↵
Fetcher, 181
- skdaccess::geo::gldas::data_fetcher::DataFetcher,
198
- skdaccess::geo::grace::data_fetcher::DataFetcher,
269
- skdaccess::geo::grace::mascon::cache::data_↵
fetcher::DataFetcher, 327
- skdaccess::geo::groundwater::data_fetcher::Data_↵
Fetcher, 132
- skdaccess::geo::imsdnhs::data_fetcher::Data_↵
Fetcher, 305
- skdaccess::geo::mahali::rinex::data_fetcher::Data_↵
Fetcher, 191
- skdaccess::geo::mahali::tec::data_fetcher::Data_↵
Fetcher, 207
- skdaccess::geo::mahali::temperature::data_fetcher_↵
::DataFetcher, 172
- skdaccess::geo::modis::cache::data_fetcher::Data_↵
Fetcher, 240
- skdaccess::geo::modis::stream::data_fetcher::↵
DataFetcher, 317
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
230
- end_hour
 - skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 251
 - skdaccess::geo::wyoming_sounding::stream::data_↵
_fetcher::DataFetcher, 155
- end_time
 - skdaccess::geo::magnetometer::data_fetcher::↵
DataFetcher, 214
- end_url
 - skdaccess::astro::tess::data::cache::DataFetcher,
298
 - skdaccess::astro::tess::simulated::cache::Data_↵
Fetcher, 114
- engineering/la/generic/stream.py, 426
- engineering/la/traffic_counts/stream.py, 426
- engineering/webcam/mit_sailing/stream.py, 425
- error_names
 - skdaccess::framework::data_class::SeriesDictionary_↵
Wrapper, 396
 - skdaccess::framework::data_class::SeriesWrapper,
402
- examples/terminal_groundwater_example.py, 427
- field_names
 - skdaccess::astro::voyager::data_fetcher::Data_↵
Fetcher, 296
- field_widths
 - skdaccess::astro::voyager::data_fetcher::Data_↵
Fetcher, 296
- file_name
 - skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 284

- files
 - skdaccess::utilities::file_browser::FileBrowser, 378
- finance/timeseries/stream.py, 425
- find_data
 - skdaccess::geo::modis::cache::data_fetcher::Data←
Fetcher, 235
- flip_y
 - skdaccess::utilities::image_util::LinearGeolocation,
389
- framework/data_class.py, 428
- framework/param_class.py, 428
- fullIDF
 - terminal_groundwater_example, 60
- fullIDW
 - terminal_groundwater_example, 60
- generate_links
 - skdaccess::geo::mahali::rinex::data_fetcher::Data←
Fetcher, 191
- generateQueries
 - skdaccess::utilities::sounding_util, 54
- generateURLFromTID
 - skdaccess::astro::tess::data::cache::DataFetcher,
298
 - skdaccess::astro::tess::generic::cache::DataFetcher,
119
 - skdaccess::astro::tess::simulated::cache::Data←
Fetcher, 114
- generateURL
 - skdaccess::astro::voyager::data_fetcher::Data←
Fetcher, 290
- geo/era_interim/cache/data_fetcher.py, 429
- geo/gldas/data_fetcher.py, 431
- geo/grace/data_fetcher.py, 432
- geo/grace/mascon/cache/data_fetcher.py, 433
- geo/groundwater/data_fetcher.py, 433
- geo/imsdnhs/data_fetcher.py, 436
- geo/magnetometer/data_fetcher.py, 435
- geo/mahali/rinex/data_fetcher.py, 434
- geo/mahali/rinex/data_wrapper.py, 437
- geo/mahali/tec/data_fetcher.py, 434
- geo/mahali/temperature/data_fetcher.py, 434
- geo/modis/cache/cloud_mask/data_fetcher.py, 430
- geo/modis/cache/cloud_opacity/data_fetcher.py, 430
- geo/modis/cache/data_fetcher.py, 429
- geo/modis/cache/reflectance/data_fetcher.py, 429
- geo/modis/stream/cloud_mask/data_fetcher.py, 431
- geo/modis/stream/cloud_opacity/data_fetcher.py, 431
- geo/modis/stream/data_fetcher.py, 431
- geo/modis/stream/reflectance/data_fetcher.py, 430
- geo/ngl_gps/data_fetcher.py, 435
- geo/pbo/data_fetcher.py, 432
- geo/sentinel_1/cache/data_fetcher.py, 433
- geo/srtm/cache/data_fetcher.py, 433
- geo/uavsar/cache/data_fetcher.py, 432
- geo/wyoming_sounding/cache/data_fetcher.py, 435
- geo/wyoming_sounding/stream/data_fetcher.py, 435
- get
 - skdaccess::framework::data_class::DataWrapper←
Base, 373
 - skdaccess::framework::data_class::ImageWrapper,
380
 - skdaccess::framework::data_class::SeriesDictionary←
Wrapper, 393
 - skdaccess::framework::data_class::SeriesWrapper,
399
 - skdaccess::framework::data_class::TableWrapper,
413
 - skdaccess::framework::data_class::XArrayWrapper,
420
 - skdaccess::geo::mahali::rinex::data_wrapper::Data←
Wrapper, 368
- get_files_urls
 - skdaccess::utilities::ode_util, 46
- get_query_url
 - skdaccess::utilities::ode_util, 47
- get_raster_array
 - skdaccess::utilities::ode_util, 47
- get_raster_extent
 - skdaccess::utilities::ode_util, 47
- getAllOptions
 - skdaccess::framework::param_class::AutoList, 68
 - skdaccess::framework::param_class::AutoListCycle,
72
 - skdaccess::framework::param_class::AutoList←
Permute, 76
 - skdaccess::framework::param_class::AutoList←
Remove, 81
 - skdaccess::framework::param_class::AutoList←
Subset, 85
- getAntennaLogs
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
225
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 256
- getConfig
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
137
 - skdaccess::astro::spectra::stream::DataFetcher, 307
 - skdaccess::astro::tess::generic::cache::DataFetcher,
120
 - skdaccess::astro::voyager::data_fetcher::Data←
Fetcher, 290
 - skdaccess::engineering::la::generic::stream::Data←
Fetcher, 143
 - skdaccess::engineering::webcam::mit_sailing←
::stream::DataFetcher, 271
 - skdaccess::finance::timeseries::stream::Data←
Fetcher, 177

- skdaccess::framework::data_class::DataFetcher↔
Base, 338
- skdaccess::framework::data_class::DataFetcher↔
Cache, 345
- skdaccess::framework::data_class::DataFetcher↔
Local, 351
- skdaccess::framework::data_class::DataFetcher↔
Storage, 357
- skdaccess::framework::data_class::DataFetcher↔
Stream, 363
- skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 107
- skdaccess::geo::gldas::data_fetcher::DataFetcher,
194
- skdaccess::geo::grace::data_fetcher::DataFetcher,
265
- skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 322
- skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 128
- skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 301
- skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 210
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 187
- skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 203
- skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 168
- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 236
- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 313
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
226
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 256
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 331
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 159
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 218
- skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 246
- skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, 150
- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 279
- skdaccess::solar::sdo::data_fetcher::DataFetcher,
100
- getConfigItem
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
137
 - skdaccess::astro::spectra::stream::DataFetcher, 308
 - skdaccess::astro::tess::generic::cache::DataFetcher,
120
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 290
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, 143
 - skdaccess::engineering::webcam::mit_sailing↔
::stream::DataFetcher, 271
 - skdaccess::finance::timeseries::stream::Data↔
Fetcher, 177
 - skdaccess::framework::data_class::DataFetcher↔
Base, 338
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 345
 - skdaccess::framework::data_class::DataFetcher↔
Local, 351
 - skdaccess::framework::data_class::DataFetcher↔
Storage, 357
 - skdaccess::framework::data_class::DataFetcher↔
Stream, 363
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 107
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
194
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
265
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 322
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 128
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 302
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 210
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 187
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 203
 - skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 168
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 236
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 314
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
226
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 256
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 331
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 159
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 218

- skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 246
- skdaccess::geo::wyoming_sounding::stream::data_↵
_fetcher::DataFetcher, 151
- skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 279
- skdaccess::solar::sdo::data_fetcher::DataFetcher,
100
- getDataLocation
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
137
 - skdaccess::astro::tess::generic::cache::DataFetcher,
120
 - skdaccess::astro::voyager::data_fetcher::Data↵
Fetcher, 291
 - skdaccess::framework::data_class::DataFetcher↵
Cache, 346
 - skdaccess::framework::data_class::DataFetcher↵
Local, 351
 - skdaccess::framework::data_class::DataFetcher↵
Storage, 357
 - skdaccess::geo::era_interim::cache::data_fetcher::↵
DataFetcher, 108
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
195
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
265
 - skdaccess::geo::grace::mascon::cache::data_↵
fetcher::DataFetcher, 323
 - skdaccess::geo::groundwater::data_fetcher::Data↵
Fetcher, 128
 - skdaccess::geo::imsdnhs::data_fetcher::Data↵
Fetcher, 302
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↵
Fetcher, 187
 - skdaccess::geo::mahali::tec::data_fetcher::Data↵
Fetcher, 204
 - skdaccess::geo::modis::cache::data_fetcher::Data↵
Fetcher, 237
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
226
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 257
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↵
DataFetcher, 332
 - skdaccess::geo::srtm::cache::data_fetcher::Data↵
Fetcher, 160
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↵
Fetcher, 219
 - skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 247
 - skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 279
- getDataMetadata
 - skdaccess::geo::magnetometer::data_fetcher::↵
DataFetcher, 210
- getDefaultColumns
 - skdaccess::framework::data_class::TableWrapper,
413
- getDefaultErrorColumns
 - skdaccess::framework::data_class::TableWrapper,
413
- getExtents
 - skdaccess::utilities::image_util::LinearGeolocation,
388
- getExtentsFromCentersPlateCarree
 - skdaccess::utilities::image_util, 37
- getFileIDs
 - skdaccess::utilities::modis_util, 42
- getFileURLs
 - skdaccess::utilities::modis_util, 43
- getGeoTransform
 - skdaccess::utilities::image_util, 37
- getHDFSStorage
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
138
 - skdaccess::astro::tess::generic::cache::DataFetcher,
121
 - skdaccess::astro::voyager::data_fetcher::Data↵
Fetcher, 291
 - skdaccess::framework::data_class::DataFetcher↵
Cache, 346
 - skdaccess::geo::era_interim::cache::data_fetcher::↵
DataFetcher, 108
 - skdaccess::geo::grace::mascon::cache::data_↵
fetcher::DataFetcher, 323
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↵
Fetcher, 188
 - skdaccess::geo::mahali::tec::data_fetcher::Data↵
Fetcher, 204
 - skdaccess::geo::modis::cache::data_fetcher::Data↵
Fetcher, 237
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↵
DataFetcher, 332
 - skdaccess::geo::srtm::cache::data_fetcher::Data↵
Fetcher, 160
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↵
Fetcher, 219
 - skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 247
 - skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 281
- getImageType
 - skdaccess::utilities::modis_util, 43
- getIndices
 - skdaccess::framework::data_class::SeriesDictionary↵
Wrapper, 393
 - skdaccess::framework::data_class::SeriesWrapper,
399

- getInfo
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 257
- getIterator
 - skdaccess::framework::data_class::DataWrapper↔
Base, 374
 - skdaccess::framework::data_class::ImageWrapper,
381
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 393
 - skdaccess::framework::data_class::SeriesWrapper,
400
 - skdaccess::framework::data_class::TableWrapper,
413
 - skdaccess::framework::data_class::XArrayWrapper,
420
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 368
- getLatLon
 - skdaccess::utilities::image_util::LinearGeolocation,
388
- getLatLonRange
 - skdaccess::utilities::pbo_util, 49
- getLength
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 393
 - skdaccess::framework::data_class::SeriesWrapper,
400
 - skdaccess::framework::data_class::TableWrapper,
414
- getMasconPlacement
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 323
- getMetadata
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
138
 - skdaccess::astro::spectra::stream::DataFetcher, 308
 - skdaccess::astro::tess::generic::cache::DataFetcher,
121
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 291
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, 144
 - skdaccess::engineering::webcam::mit_sailing↔
::stream::DataFetcher, 272
 - skdaccess::finance::timeseries::stream::Data↔
Fetcher, 178
 - skdaccess::framework::data_class::DataFetcher↔
Base, 339
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 346
 - skdaccess::framework::data_class::DataFetcher↔
Local, 352
 - skdaccess::framework::data_class::DataFetcher↔
Storage, 358
 - skdaccess::framework::data_class::DataFetcher↔
Stream, 363
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 109
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
195
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
266
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 324
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 129
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 302
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 211
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 188
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 204
 - skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 169
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 237
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 314
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
227
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 257
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 332
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 160
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 219
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 247
 - skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, 151
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 281
 - skdaccess::solar::sdo::data_fetcher::DataFetcher,
101
- getMetadataFiles
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 292
- getModisData
 - skdaccess::utilities::modis_util, 44
- getPixelYX
 - skdaccess::utilities::image_util::AffineGlobalCoords,
64
- getProjectedYX
 - skdaccess::utilities::image_util::AffineGlobalCoords,
64

- getROIstations
 - skdaccess::utilities::pbo_util, [50](#)
- getResults
 - skdaccess::framework::data_class::DataWrapper↔
Base, [374](#)
 - skdaccess::framework::data_class::ImageWrapper,
[381](#)
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, [394](#)
 - skdaccess::framework::data_class::SeriesWrapper,
[400](#)
 - skdaccess::framework::data_class::TableWrapper,
[414](#)
 - skdaccess::framework::data_class::XArrayWrapper,
[420](#)
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, [369](#)
- getRunID
 - skdaccess::framework::data_class::DataWrapper↔
Base, [374](#)
 - skdaccess::framework::data_class::ImageWrapper,
[381](#)
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, [394](#)
 - skdaccess::framework::data_class::SeriesWrapper,
[400](#)
 - skdaccess::framework::data_class::TableWrapper,
[414](#)
 - skdaccess::framework::data_class::XArrayWrapper,
[420](#)
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, [369](#)
- getSRTMData
 - skdaccess::utilities::srtm_util, [55](#)
- getSRTMLatLon
 - skdaccess::utilities::srtm_util, [56](#)
- getStartEndDate
 - skdaccess::utilities::grace_util, [33](#)
- getStationCoords
 - skdaccess::utilities::pbo_util, [50](#)
- getStationMetadata
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, [129](#)
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
[227](#)
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, [257](#)
- getTargetInformation
 - skdaccess::astro::tess::data::cache::DataFetcher,
[298](#)
 - skdaccess::astro::tess::generic::cache::DataFetcher,
[121](#)
 - skdaccess::astro::tess::simulated::cache::Data↔
Fetcher, [114](#)
- getYX
 - skdaccess::utilities::image_util::LinearGeolocation,
[389](#)
- grid
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [240](#)
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, [317](#)
- grid_fill
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [240](#)
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, [317](#)
- handle_data
 - skdaccess::utilities::sounding_util::SoundingParser,
[404](#)
- handle_endtag
 - skdaccess::utilities::sounding_util::SoundingParser,
[405](#)
- handle_starttag
 - skdaccess::utilities::sounding_util::SoundingParser,
[405](#)
- in_header
 - skdaccess::utilities::sounding_util::SoundingParser,
[405](#)
- in_pre_tag
 - skdaccess::utilities::sounding_util::SoundingParser,
[406](#)
- index
 - skdaccess::framework::param_class::AutoListCycle,
[73](#)
- index_date_only
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, [261](#)
- index_list
 - skdaccess::framework::data_class::XArrayWrapper,
[422](#)
- info
 - skdaccess::framework::data_class::DataWrapper↔
Base, [374](#)
 - skdaccess::framework::data_class::ImageWrapper,
[381](#)
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, [394](#)
 - skdaccess::framework::data_class::SeriesWrapper,
[401](#)
 - skdaccess::framework::data_class::TableWrapper,
[414](#)
 - skdaccess::framework::data_class::XArrayWrapper,
[421](#)
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, [369](#)
- instrument
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [284](#)

- interval
 - skdaccess::finance::timeseries::stream::Data↔
Fetcher, 181
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 214
- join_string
 - skdaccess::utilities::support, 57
- label
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, 147
 - skdaccess::utilities::sounding_util::SoundingParser,
406
- label_1
 - terminal_groundwater_example, 61
- label_2
 - terminal_groundwater_example, 61
- lat_data
 - skdaccess::utilities::modis_util::LatLon, 386
- lat_extents
 - skdaccess::utilities::image_util::LinearGeolocation,
389
- lat_func
 - skdaccess::utilities::image_util::SplineLatLon, 409
- lat_pixel_size
 - skdaccess::utilities::image_util::LinearGeolocation,
389
- lat_range
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
230
- lat_spline
 - skdaccess::utilities::image_util, 38
- lat_tile_end
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 164
- lat_tile_start
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 164
- len_x
 - skdaccess::utilities::image_util::LinearGeolocation,
390
- len_y
 - skdaccess::utilities::image_util::LinearGeolocation,
390
- list_val_list
 - skdaccess::framework::param_class::AutoListCycle,
73
- llh_url
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 222
- local_paths
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 335
- lon_data
 - skdaccess::utilities::modis_util::LatLon, 386
- lon_extents
 - skdaccess::utilities::image_util::LinearGeolocation,
390
- lon_func
 - skdaccess::utilities::image_util::SplineLatLon, 409
- lon_pixel_size
 - skdaccess::utilities::image_util::LinearGeolocation,
390
- lon_range
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
230
- lon_spline
 - skdaccess::utilities::image_util, 38
- lon_tile_end
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 164
- lon_tile_start
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 164
- mascon_placement_url
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 327
- mascon_url
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 327
- mask_water
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 164
- max_lat
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 284
- max_ob_time
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 284
- mdyratio
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
230
- memmap
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 222
- merge_srtm_tiles
 - skdaccess::utilities::srtm_util, 56
- meta_data
 - skdaccess::framework::data_class::DataWrapper↔
Base, 376
 - skdaccess::framework::data_class::ImageWrapper,
383
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 396
 - skdaccess::framework::data_class::SeriesWrapper,
403

- skdaccess::framework::data_class::TableWrapper, 417
- skdaccess::framework::data_class::XArrayWrapper, 422
- skdaccess::geo::mahali::rinex::data_wrapper::Data↔Wrapper, 371
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 261
- terminal_groundwater_example, 61
- metadata_dict
 - skdaccess::utilities::sounding_util::SoundingParser, 406
- metadata_url_list
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔Fetcher, 222
- min_lat
 - skdaccess::planetary::ode::cache::data_fetcher::↔DataFetcher, 285
- min_ob_time
 - skdaccess::planetary::ode::cache::data_fetcher::↔DataFetcher, 285
- mission
 - skdaccess::planetary::ode::cache::data_fetcher::↔DataFetcher, 285
- modis_id
 - skdaccess::geo::modis::cache::data_fetcher::Data↔Fetcher, 240
 - skdaccess::geo::modis::stream::data_fetcher::↔DataFetcher, 318
- modis_identifier
 - skdaccess::geo::modis::cache::data_fetcher::Data↔Fetcher, 241
 - skdaccess::geo::modis::stream::data_fetcher::↔DataFetcher, 318
- modis_platform
 - skdaccess::geo::modis::cache::data_fetcher::Data↔Fetcher, 241
 - skdaccess::geo::modis::stream::data_fetcher::↔DataFetcher, 318
- month_list
 - skdaccess::geo::wyoming_sounding::cache::data_↔fetcher::DataFetcher, 251
 - skdaccess::geo::wyoming_sounding::stream::data_↔fetcher::DataFetcher, 155
- multirun_enabled
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 138
 - skdaccess::astro::spectra::stream::DataFetcher, 308
 - skdaccess::astro::tess::generic::cache::DataFetcher, 122
 - skdaccess::astro::voyager::data_fetcher::Data↔Fetcher, 292
 - skdaccess::engineering::la::generic::stream::Data↔Fetcher, 144
 - skdaccess::engineering::webcam::mit_sailing↔↔stream::DataFetcher, 272
- skdaccess::finance::timeseries::stream::Data↔Fetcher, 178
- skdaccess::framework::data_class::DataFetcher↔Base, 339
- skdaccess::framework::data_class::DataFetcher↔Cache, 347
- skdaccess::framework::data_class::DataFetcher↔Local, 352
- skdaccess::framework::data_class::DataFetcher↔Storage, 358
- skdaccess::framework::data_class::DataFetcher↔Stream, 364
- skdaccess::geo::era_interim::cache::data_fetcher::↔DataFetcher, 109
- skdaccess::geo::gldas::data_fetcher::DataFetcher, 195
- skdaccess::geo::grace::data_fetcher::DataFetcher, 266
- skdaccess::geo::grace::mascon::cache::data_↔fetcher::DataFetcher, 324
- skdaccess::geo::groundwater::data_fetcher::Data↔Fetcher, 129
- skdaccess::geo::imsdnhs::data_fetcher::Data↔Fetcher, 303
- skdaccess::geo::magnetometer::data_fetcher::↔DataFetcher, 211
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔Fetcher, 188
- skdaccess::geo::mahali::tec::data_fetcher::Data↔Fetcher, 205
- skdaccess::geo::mahali::temperature::data_fetcher↔↔DataFetcher, 169
- skdaccess::geo::modis::cache::data_fetcher::Data↔Fetcher, 238
- skdaccess::geo::modis::stream::data_fetcher::↔DataFetcher, 314
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 227
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 258
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔DataFetcher, 333
- skdaccess::geo::srtm::cache::data_fetcher::Data↔Fetcher, 161
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔Fetcher, 220
- skdaccess::geo::wyoming_sounding::cache::data_↔fetcher::DataFetcher, 248
- skdaccess::geo::wyoming_sounding::stream::data_↔fetcher::DataFetcher, 151
- skdaccess::planetary::ode::cache::data_fetcher::↔DataFetcher, 281
- skdaccess::solar::sdo::data_fetcher::DataFetcher, 101

- n
 - skdaccess::framework::param_class::AutoList↔
Remove, [82](#)
 - skdaccess::framework::param_class::AutoParam↔
MinMax, [97](#)
- n_max
 - skdaccess::framework::param_class::AutoParam↔
MinMax, [97](#)
- normalize
 - skdaccess::utilities::kepler_util, [39](#)
- nostab_sys
 - skdaccess::utilities::pbo_util, [51](#)
- number_product_limit
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [285](#)
- openPandasHDFStoreLocking
 - skdaccess::utilities::file_util, [31](#)
- output
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
[139](#)
 - skdaccess::astro::spectra::stream::DataFetcher, [309](#)
 - skdaccess::astro::tess::generic::cache::DataFetcher,
[122](#)
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [292](#)
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, [144](#)
 - skdaccess::engineering::webcam::mit_sailing↔
::stream::DataFetcher, [272](#)
 - skdaccess::finance::timeseries::stream::Data↔
Fetcher, [178](#)
 - skdaccess::framework::data_class::DataFetcher↔
Base, [339](#)
 - skdaccess::framework::data_class::DataFetcher↔
Cache, [347](#)
 - skdaccess::framework::data_class::DataFetcher↔
Local, [352](#)
 - skdaccess::framework::data_class::DataFetcher↔
Storage, [358](#)
 - skdaccess::framework::data_class::DataFetcher↔
Stream, [364](#)
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, [109](#)
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
[196](#)
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
[266](#)
 - skdaccess::geo::grace::mascon::cache::data↔
fetcher::DataFetcher, [324](#)
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, [129](#)
 - skdaccess::geo::imdsnhs::data_fetcher::Data↔
Fetcher, [303](#)
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, [211](#)
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, [189](#)
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, [205](#)
 - skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, [169](#)
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [238](#)
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, [315](#)
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
[228](#)
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, [258](#)
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [333](#)
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [161](#)
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, [220](#)
 - skdaccess::geo::wyoming_sounding::cache::data↔
fetcher::DataFetcher, [248](#)
 - skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, [152](#)
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [282](#)
 - skdaccess::solar::sdo::data_fetcher::DataFetcher,
[101](#)
- pandas_kwargs
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, [147](#)
- parameters
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, [148](#)
- parselonoFile
 - skdaccess::utilities::mahali_util, [40](#)
- parseSatelliteData
 - skdaccess::utilities::sentinel_1_util, [53](#)
- parseTessData
 - skdaccess::utilities::tess_utils, [58](#)
- parseVoyagerData
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [293](#)
- parseVoyagerMetadata
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [293](#)
- password
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, [112](#)
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [335](#)

- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [164](#)
- path
 - skdaccess::utilities::file_browser::FileBrowser, [378](#)
- perturb
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, [139](#)
 - skdaccess::astro::spectra::stream::DataFetcher, [309](#)
 - skdaccess::astro::tess::generic::cache::DataFetcher, [122](#)
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [294](#)
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, [145](#)
 - skdaccess::engineering::webcam::mit_sailing↔
::stream::DataFetcher, [272](#)
 - skdaccess::finance::timeseries::stream::Data↔
Fetcher, [178](#)
 - skdaccess::framework::data_class::DataFetcher↔
Base, [340](#)
 - skdaccess::framework::data_class::DataFetcher↔
Cache, [347](#)
 - skdaccess::framework::data_class::DataFetcher↔
Local, [352](#)
 - skdaccess::framework::data_class::DataFetcher↔
Storage, [359](#)
 - skdaccess::framework::data_class::DataFetcher↔
Stream, [364](#)
 - skdaccess::framework::param_class::AutoList, [68](#)
 - skdaccess::framework::param_class::AutoListCycle, [72](#)
 - skdaccess::framework::param_class::AutoList↔
Permute, [77](#)
 - skdaccess::framework::param_class::AutoList↔
Remove, [81](#)
 - skdaccess::framework::param_class::AutoList↔
Subset, [85](#)
 - skdaccess::framework::param_class::AutoParam, [88](#)
 - skdaccess::framework::param_class::AutoParamList, [91](#)
 - skdaccess::framework::param_class::AutoParam↔
ListCycle, [93](#)
 - skdaccess::framework::param_class::AutoParam↔
MinMax, [96](#)
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, [109](#)
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, [196](#)
 - skdaccess::geo::grace::data_fetcher::DataFetcher, [267](#)
 - skdaccess::geo::grace::mascon::cache::data↔
fetcher::DataFetcher, [324](#)
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, [130](#)
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, [303](#)
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, [211](#)
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, [189](#)
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, [205](#)
 - skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, [170](#)
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [238](#)
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, [315](#)
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, [228](#)
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, [258](#)
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [333](#)
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [161](#)
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, [220](#)
 - skdaccess::geo::wyoming_sounding::cache::data↔
fetcher::DataFetcher, [248](#)
 - skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, [152](#)
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [282](#)
 - skdaccess::solar::sdo::data_fetcher::DataFetcher, [101](#)
 - planetary/ode/cache/data_fetcher.py, [437](#)
 - polarization
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [336](#)
 - possible_data_types
 - skdaccess::finance::timeseries::stream::Data↔
Fetcher, [181](#)
 - possible_intervals
 - skdaccess::finance::timeseries::stream::Data↔
Fetcher, [181](#)
 - product_id
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [285](#)
 - product_type
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [285](#)
 - progress_bar
 - skdaccess::utilities::support, [57](#)
 - propagateErrors
 - skdaccess::utilities::pbo_util, [52](#)
 - quarter_list

- skdaccess::astro::kepler::data_fetcher::DataFetcher, 141
- query_files_urls
 - skdaccess::utilities::ode_util, 48
- query_yes_no
 - skdaccess::utilities::ode_util, 49
- read_data
 - skdaccess::utilities::sounding_util::SoundingParser, 406
- readMODISData
 - skdaccess::utilities::modis_util, 44
- readTellusData
 - skdaccess::utilities::grace_util, 33
- readUAVSARMetadata
 - skdaccess::utilities::uavsar_util, 59
- remove_ndv
 - skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 285
- removeAntennaOffset
 - skdaccess::utilities::pbo_util, 52
- removeFrames
 - skdaccess::framework::data_class::TableWrapper, 415
- resample
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 198
- rescale
 - skdaccess::utilities::modis_util, 45
- reset
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 139
 - skdaccess::astro::spectra::stream::DataFetcher, 309
 - skdaccess::astro::tess::generic::cache::DataFetcher, 122
 - skdaccess::astro::voyager::data_fetcher::DataFetcher, 294
 - skdaccess::engineering::la::generic::stream::DataFetcher, 145
 - skdaccess::engineering::webcam::mit_sailing::stream::DataFetcher, 273
 - skdaccess::finance::timeseries::stream::DataFetcher, 179
 - skdaccess::framework::data_class::DataFetcherBase, 340
 - skdaccess::framework::data_class::DataFetcherCache, 347
 - skdaccess::framework::data_class::DataFetcherLocal, 353
 - skdaccess::framework::data_class::DataFetcherStorage, 359
 - skdaccess::framework::data_class::DataFetcherStream, 364
 - skdaccess::framework::data_class::DataWrapperBase, 375
 - skdaccess::framework::data_class::ImageWrapper, 382
 - skdaccess::framework::data_class::SeriesDictionaryWrapper, 394
 - skdaccess::framework::data_class::SeriesWrapper, 401
 - skdaccess::framework::data_class::TableWrapper, 415
 - skdaccess::framework::data_class::XArrayWrapper, 421
 - skdaccess::framework::param_class::AutoList, 68
 - skdaccess::framework::param_class::AutoListCycle, 73
 - skdaccess::framework::param_class::AutoListPermute, 77
 - skdaccess::framework::param_class::AutoListRemove, 81
 - skdaccess::framework::param_class::AutoListSubset, 85
 - skdaccess::framework::param_class::AutoParam, 88
 - skdaccess::framework::param_class::AutoParamList, 91
 - skdaccess::framework::param_class::AutoParamListCycle, 94
 - skdaccess::framework::param_class::AutoParamMinMax, 97
 - skdaccess::geo::era_interim::cache::data_fetcher::DataFetcher, 110
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 196
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 267
 - skdaccess::geo::grace::mascon::cache::data_fetcher::DataFetcher, 325
 - skdaccess::geo::groundwater::data_fetcher::DataFetcher, 130
 - skdaccess::geo::imsdnhs::data_fetcher::DataFetcher, 303
 - skdaccess::geo::magnetometer::data_fetcher::DataFetcher, 212
 - skdaccess::geo::mahali::rinex::data_fetcher::DataFetcher, 189
 - skdaccess::geo::mahali::rinex::data_wrapper::DataWrapper, 369
 - skdaccess::geo::mahali::tec::data_fetcher::DataFetcher, 205
 - skdaccess::geo::mahali::temperature::data_fetcher::DataFetcher, 170
 - skdaccess::geo::modis::cache::data_fetcher::DataFetcher, 238
 - skdaccess::geo::modis::stream::data_fetcher::DataFetcher, 315

- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 228
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 258
- skdaccess::geo::sentinel_1::cache::data_fetcher::DataFetcher, 333
- skdaccess::geo::srtm::cache::data_fetcher::DataFetcher, 161
- skdaccess::geo::uavsar::cache::data_fetcher::DataFetcher, 220
- skdaccess::geo::wyoming_sounding::cache::data_fetcher::DataFetcher, 248
- skdaccess::geo::wyoming_sounding::stream::data_fetcher::DataFetcher, 152
- skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 282
- skdaccess::solar::sdo::data_fetcher::DataFetcher, 102
- result_offset_number
 - skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 286
- results
 - skdaccess::framework::data_class::DataWrapperBase, 376
 - skdaccess::framework::data_class::ImageWrapper, 383
 - skdaccess::framework::data_class::SeriesDictionaryWrapper, 396
 - skdaccess::framework::data_class::SeriesWrapper, 403
 - skdaccess::framework::data_class::TableWrapper, 417
 - skdaccess::framework::data_class::XArrayWrapper, 423
 - skdaccess::geo::mahali::rinex::data_wrapper::DataWrapper, 371
- retrieveCommonDatesHDF
 - skdaccess::utilities::support, 58
- retrieveOnlineData
 - skdaccess::astro::spectra::stream::DataFetcher, 309
 - skdaccess::engineering::la::generic::stream::DataFetcher, 145
 - skdaccess::engineering::webcam::mit_sailing::stream::DataFetcher, 273
 - skdaccess::finance::timeseries::stream::DataFetcher, 179
 - skdaccess::framework::data_class::DataFetcherLocal, 353
 - skdaccess::framework::data_class::DataFetcherStorage, 359
 - skdaccess::geo::era_interim::cache::data_fetcher::DataFetcher, 110
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 196
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 267
 - skdaccess::geo::grace::mascon::cache::data_fetcher::DataFetcher, 325
 - skdaccess::geo::groundwater::data_fetcher::DataFetcher, 130
 - skdaccess::geo::imsdnhs::data_fetcher::DataFetcher, 303
 - skdaccess::geo::mahali::rinex::data_fetcher::DataFetcher, 189
 - skdaccess::geo::mahali::tec::data_fetcher::DataFetcher, 205
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 228
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 258
 - skdaccess::geo::sentinel_1::cache::data_fetcher::DataFetcher, 333
 - skdaccess::geo::srtm::cache::data_fetcher::DataFetcher, 161
 - skdaccess::geo::uavsar::cache::data_fetcher::DataFetcher, 220
 - skdaccess::geo::wyoming_sounding::cache::data_fetcher::DataFetcher, 248
 - skdaccess::geo::wyoming_sounding::stream::data_fetcher::DataFetcher, 152
 - skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 282
 - skdaccess::solar::sdo::data_fetcher::DataFetcher, 102
 - run_id
 - skdaccess::framework::data_class::DataWrapperBase, 376
 - skdaccess::framework::data_class::ImageWrapper, 384
 - skdaccess::framework::data_class::SeriesDictionaryWrapper, 396
 - skdaccess::framework::data_class::SeriesWrapper, 403
 - skdaccess::framework::data_class::TableWrapper, 418
 - skdaccess::framework::data_class::XArrayWrapper, 423
 - skdaccess::geo::mahali::rinex::data_wrapper::DataWrapper, 371
 - satellite_url_list
 - skdaccess::geo::sentinel_1::cache::data_fetcher::DataFetcher, 336
 - scale_factor_url
 - skdaccess::geo::grace::mascon::cache::data_fetcher::DataFetcher, 327
 - setDataLocation
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 139
 - skdaccess::astro::tess::generic::cache::DataFetcher, 123
 - skdaccess::astro::voyager::data_fetcher::DataFetcher, 294
 - skdaccess::framework::data_class::DataFetcherCache, 347
 - skdaccess::framework::data_class::DataFetcherLocal, 353
 - skdaccess::framework::data_class::DataFetcherStorage, 359
 - skdaccess::geo::era_interim::cache::data_fetcher::DataFetcher, 110
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 196
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 267
 - skdaccess::geo::grace::mascon::cache::data_fetcher::DataFetcher, 325
 - skdaccess::geo::groundwater::data_fetcher::DataFetcher, 130
 - skdaccess::geo::imsdnhs::data_fetcher::DataFetcher, 303
 - skdaccess::geo::mahali::rinex::data_fetcher::DataFetcher, 189
 - skdaccess::geo::mahali::tec::data_fetcher::DataFetcher, 205
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 228
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 258
 - skdaccess::geo::sentinel_1::cache::data_fetcher::DataFetcher, 336
 - skdaccess::geo::srtm::cache::data_fetcher::DataFetcher, 161
 - skdaccess::geo::uavsar::cache::data_fetcher::DataFetcher, 220
 - skdaccess::geo::wyoming_sounding::cache::data_fetcher::DataFetcher, 248
 - skdaccess::geo::wyoming_sounding::stream::data_fetcher::DataFetcher, 152
 - skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 282
 - skdaccess::solar::sdo::data_fetcher::DataFetcher, 102

- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 238
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
228
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 259
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 334
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 162
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 220
- skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 249
- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 282
- setStationList
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 259
- skdaccess, 13
- skdaccess.astro, 13
- skdaccess.astro.kepler, 13
- skdaccess.astro.kepler.data_fetcher, 14
- skdaccess.astro.kepler.DataFetcher, 133
- skdaccess.astro.spectra, 14
- skdaccess.astro.spectra.stream, 14
- skdaccess.astro.spectra.stream.DataFetcher, 306
- skdaccess.astro.tess, 14
- skdaccess.astro.tess.data, 14
- skdaccess.astro.tess.data.cache, 14
- skdaccess.astro.tess.data.cache.DataFetcher, 297
- skdaccess.astro.tess.generic, 15
- skdaccess.astro.tess.generic.cache, 15
- skdaccess.astro.tess.generic.cache.DataFetcher, 116
- skdaccess.astro.tess.simulated, 15
- skdaccess.astro.tess.simulated.cache, 15
- skdaccess.astro.tess.simulated.cache.DataFetcher, 113
- skdaccess.astro.voyager, 15
- skdaccess.astro.voyager.data_fetcher, 15
- skdaccess.astro.voyager.DataFetcher, 286
- skdaccess.engineering, 16
- skdaccess.engineering.la, 16
- skdaccess.engineering.la.generic, 16
- skdaccess.engineering.la.generic.stream, 16
- skdaccess.engineering.la.generic.stream.DataFetcher,
141
- skdaccess.engineering.la.traffic_counts, 16
- skdaccess.engineering.la.traffic_counts.stream, 16
- skdaccess.engineering.la.traffic_counts.stream.Data↔
Fetcher, 199
- skdaccess.engineering.webcam, 17
- skdaccess.engineering.webcam.mit_sailing, 17
- skdaccess.engineering.webcam.mit_sailing.stream, 17
- skdaccess.engineering.webcam.mit_sailing.stream.↔
DataFetcher, 269
- skdaccess.finance, 17
- skdaccess.finance.timeseries, 17
- skdaccess.finance.timeseries.stream, 17
- skdaccess.finance.timeseries.stream.DataFetcher, 175
- skdaccess.framework, 18
- skdaccess.framework.data_class, 18
- skdaccess.framework.data_class.DataFetcherBase, 337
- skdaccess.framework.data_class.DataFetcherCache, 342
- skdaccess.framework.data_class.DataFetcherLocal, 349
- skdaccess.framework.data_class.DataFetcherStorage,
355
- skdaccess.framework.data_class.DataFetcherStream,
361
- skdaccess.framework.data_class.DataWrapperBase, 371
- skdaccess.framework.data_class.ImageWrapper, 378
- skdaccess.framework.data_class.SeriesDictionary↔
Wrapper, 391
- skdaccess.framework.data_class.SeriesWrapper, 397
- skdaccess.framework.data_class.TableWrapper, 410
- skdaccess.framework.data_class.XArrayWrapper, 418
- skdaccess.framework.param_class, 18
- skdaccess.framework.param_class.AutoList, 65
- skdaccess.framework.param_class.AutoListCycle, 69
- skdaccess.framework.param_class.AutoListPermute, 74
- skdaccess.framework.param_class.AutoListRemove, 78
- skdaccess.framework.param_class.AutoListSubset, 82
- skdaccess.framework.param_class.AutoParam, 86
- skdaccess.framework.param_class.AutoParamList, 89
- skdaccess.framework.param_class.AutoParamListCycle,
92
- skdaccess.framework.param_class.AutoParamMinMax,
95
- skdaccess.geo, 19
- skdaccess.geo.era_interim, 19
- skdaccess.geo.era_interim.cache, 19
- skdaccess.geo.era_interim.cache.data_fetcher, 20
- skdaccess.geo.era_interim.cache.DataFetcher, 104
- skdaccess.geo.gldas, 20
- skdaccess.geo.gldas.data_fetcher, 20
- skdaccess.geo.gldas.DataFetcher, 192
- skdaccess.geo.grace, 20
- skdaccess.geo.grace.data_fetcher, 20
- skdaccess.geo.grace.DataFetcher, 262
- skdaccess.geo.grace.mascon, 20
- skdaccess.geo.grace.mascon.cache, 21
- skdaccess.geo.grace.mascon.cache.data_fetcher, 21
- skdaccess.geo.grace.mascon.cache.DataFetcher, 319
- skdaccess.geo.groundwater, 21
- skdaccess.geo.groundwater.data_fetcher, 21
- skdaccess.geo.groundwater.DataFetcher, 125
- skdaccess.geo.imsdnhs, 21
- skdaccess.geo.imsdnhs.data_fetcher, 21
- skdaccess.geo.imsdnhs.DataFetcher, 299
- skdaccess.geo.magnetometer, 22
- skdaccess.geo.magnetometer.data_fetcher, 22

- skdaccess.geo.magnetometer.DataFetcher, 208
- skdaccess.geo.mahali, 22
- skdaccess.geo.mahali.rinex, 22
- skdaccess.geo.mahali.rinex.data_fetcher, 22
- skdaccess.geo.mahali.rinex.data_wrapper, 22
- skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper, 366
- skdaccess.geo.mahali.rinex.DataFetcher, 183
- skdaccess.geo.mahali.tec, 23
- skdaccess.geo.mahali.tec.data_fetcher, 23
- skdaccess.geo.mahali.tec.DataFetcher, 200
- skdaccess.geo.mahali.temperature, 23
- skdaccess.geo.mahali.temperature.data_fetcher, 23
- skdaccess.geo.mahali.temperature.DataFetcher, 166
- skdaccess.geo.modis, 23
- skdaccess.geo.modis.cache, 23
- skdaccess.geo.modis.cache.cloud_mask, 24
- skdaccess.geo.modis.cache.cloud_mask.data_fetcher, 24
- skdaccess.geo.modis.cache.cloud_mask.DataFetcher, 182
- skdaccess.geo.modis.cache.cloud_opacity, 24
- skdaccess.geo.modis.cache.cloud_opacity.data_fetcher, 24
- skdaccess.geo.modis.cache.cloud_opacity.DataFetcher, 173
- skdaccess.geo.modis.cache.data_fetcher, 24
- skdaccess.geo.modis.cache.DataFetcher, 231
- skdaccess.geo.modis.cache.reflectance, 24
- skdaccess.geo.modis.cache.reflectance.data_fetcher, 25
- skdaccess.geo.modis.cache.reflectance.DataFetcher, 165
- skdaccess.geo.modis.stream, 25
- skdaccess.geo.modis.stream.cloud_mask, 25
- skdaccess.geo.modis.stream.cloud_mask.data_fetcher, 25
- skdaccess.geo.modis.stream.cloud_mask.DataFetcher, 174
- skdaccess.geo.modis.stream.cloud_opacity, 25
- skdaccess.geo.modis.stream.cloud_opacity.data_fetcher, 25
- skdaccess.geo.modis.stream.cloud_opacity.DataFetcher, 115
- skdaccess.geo.modis.stream.data_fetcher, 26
- skdaccess.geo.modis.stream.DataFetcher, 311
- skdaccess.geo.modis.stream.reflectance, 26
- skdaccess.geo.modis.stream.reflectance.data_fetcher, 26
- skdaccess.geo.modis.stream.reflectance.DataFetcher, 242
- skdaccess.geo.ngl_gps, 26
- skdaccess.geo.ngl_gps.data_fetcher, 26
- skdaccess.geo.ngl_gps.DataFetcher, 223
- skdaccess.geo.pbo, 26
- skdaccess.geo.pbo.data_fetcher, 27
- skdaccess.geo.pbo.DataFetcher, 252
- skdaccess.geo.sentinel_1, 27
- skdaccess.geo.sentinel_1.cache, 27
- skdaccess.geo.sentinel_1.cache.data_fetcher, 27
- skdaccess.geo.sentinel_1.cache.DataFetcher, 328
- skdaccess.geo.srtm, 27
- skdaccess.geo.srtm.cache, 27
- skdaccess.geo.srtm.cache.data_fetcher, 28
- skdaccess.geo.srtm.cache.DataFetcher, 156
- skdaccess.geo.uavsar, 28
- skdaccess.geo.uavsar.cache, 28
- skdaccess.geo.uavsar.cache.data_fetcher, 28
- skdaccess.geo.uavsar.cache.DataFetcher, 215
- skdaccess.geo.wyoming_sounding, 28
- skdaccess.geo.wyoming_sounding.cache, 28
- skdaccess.geo.wyoming_sounding.cache.data_fetcher, 29
- skdaccess.geo.wyoming_sounding.cache.DataFetcher, 243
- skdaccess.geo.wyoming_sounding.stream, 29
- skdaccess.geo.wyoming_sounding.stream.data_fetcher, 29
- skdaccess.geo.wyoming_sounding.stream.DataFetcher, 148
- skdaccess.planetary, 29
- skdaccess.planetary.ode, 29
- skdaccess.planetary.ode.cache, 29
- skdaccess.planetary.ode.cache.data_fetcher, 30
- skdaccess.planetary.ode.cache.DataFetcher, 275
- skdaccess.solar, 30
- skdaccess.solar.sdo, 30
- skdaccess.solar.sdo.data_fetcher, 30
- skdaccess.solar.sdo.DataFetcher, 98
- skdaccess.utilities, 30
- skdaccess.utilities.file_browser, 31
- skdaccess.utilities.file_browser.FileBrowser, 377
- skdaccess.utilities.file_util, 31
- skdaccess.utilities.grace_util, 31
- skdaccess.utilities.gw_util, 34
- skdaccess.utilities.image_util, 36
- skdaccess.utilities.image_util.AffineGlobalCoords, 63
- skdaccess.utilities.image_util.LinearGeolocation, 387
- skdaccess.utilities.image_util.SplineLatLon, 407
- skdaccess.utilities.kepler_util, 39
- skdaccess.utilities.mahali_util, 39
- skdaccess.utilities.modis_util, 40
- skdaccess.utilities.modis_util.LatLon, 384
- skdaccess.utilities.ode_util, 45
- skdaccess.utilities.pbo_util, 49
- skdaccess.utilities.sentinel_1_util, 53
- skdaccess.utilities.sounding_util, 54
- skdaccess.utilities.sounding_util.SoundingParser, 403
- skdaccess.utilities.srtm_util, 55
- skdaccess.utilities.support, 56
- skdaccess.utilities.tess_utils, 58
- skdaccess.utilities.uavsar_util, 59

skdaccess::astro::kepler::data_fetcher::DataFetcher

- __init__, 134
- __str__, 135
- ap_paramList, 141
- cacheData, 135
- checkIfDataExists, 136
- downloadKeplerData, 136
- getConfig, 137
- getConfigItem, 137
- getDataLocation, 137
- getHDFSStorage, 138
- getMetadata, 138
- multirun_enabled, 138
- output, 139
- perturb, 139
- quarter_list, 141
- reset, 139
- setDataLocation, 139
- verbose, 141
- verbose_print, 140
- writeConfig, 140
- writeConfigItem, 140

skdaccess::astro::spectra::stream::DataFetcher

- __init__, 307
- __str__, 307
- ap_paramList, 311
- getConfig, 307
- getConfigItem, 308
- getMetadata, 308
- multirun_enabled, 308
- output, 309
- perturb, 309
- reset, 309
- retrieveOnlineData, 309
- verbose, 311
- verbose_print, 310
- writeConfig, 310
- writeConfigItem, 310

skdaccess::astro::tess::data::cache::DataFetcher

- __init__, 297
- end_url, 298
- generateURLFromTID, 298
- getTargetInformation, 298
- start_url, 298

skdaccess::astro::tess::generic::cache::DataFetcher

- __init__, 117
- __str__, 118
- ap_paramList, 124
- cacheData, 118
- checkIfDataExists, 119
- generateURLFromTID, 119
- getConfig, 120
- getConfigItem, 120
- getDataLocation, 120

- getHDFSStorage, 121
- getMetadata, 121
- getTargetInformation, 121
- multirun_enabled, 122
- output, 122
- perturb, 122
- reset, 122
- setDataLocation, 123
- toi_information, 124
- verbose, 124
- verbose_print, 123
- writeConfig, 123
- writeConfigItem, 124

skdaccess::astro::tess::simulated::cache::DataFetcher

- __init__, 113
- end_url, 114
- generateURLFromTID, 114
- getTargetInformation, 114
- start_url, 114

skdaccess::astro::voyager::data_fetcher::DataFetcher

- __init__, 288
- __str__, 288
- ap_paramList, 296
- base_url, 296
- cacheData, 288
- checkIfDataExists, 289
- field_names, 296
- field_widths, 296
- generateURL, 290
- getConfig, 290
- getConfigItem, 290
- getDataLocation, 291
- getHDFSStorage, 291
- getMetadata, 291
- getMetadataFiles, 292
- multirun_enabled, 292
- output, 292
- parseVoyagerData, 293
- parseVoyagerMetadata, 293
- perturb, 294
- reset, 294
- setDataLocation, 294
- spacecraft_list, 296
- verbose, 296
- verbose_print, 294
- writeConfig, 295
- writeConfigItem, 295
- year_list, 296

skdaccess::engineering::la::generic::stream::DataFetcher

- __init__, 143
- __str__, 143
- ap_paramList, 147
- app_token, 147
- base_url, 147

- base_url_and_endpoint, 147
- getConfig, 143
- getConfigItem, 143
- getMetadata, 144
- label, 147
- multirun_enabled, 144
- output, 144
- pandas_kwargs, 147
- parameters, 148
- perturb, 145
- reset, 145
- retrieveOnlineData, 145
- verbose, 148
- verbose_print, 146
- writeConfig, 146
- writeConfigItem, 146
- skdaccess::engineering::la::traffic_counts::stream::DataFetcher
 - __init__, 199
- skdaccess::engineering::webcam::mit_sailing::stream::DataFetcher
 - __init__, 270
 - __str__, 271
 - ap_paramList, 274
 - camera_list, 275
 - getConfig, 271
 - getConfigItem, 271
 - getMetadata, 272
 - multirun_enabled, 272
 - output, 272
 - perturb, 272
 - reset, 273
 - retrieveOnlineData, 273
 - verbose, 275
 - verbose_print, 273
 - writeConfig, 274
 - writeConfigItem, 274
- skdaccess::finance::timeseries::stream::DataFetcher
 - __init__, 176
 - __str__, 177
 - ap_paramList, 180
 - data_type, 181
 - end_date, 181
 - getConfig, 177
 - getConfigItem, 177
 - getMetadata, 178
 - interval, 181
 - multirun_enabled, 178
 - output, 178
 - perturb, 178
 - possible_data_types, 181
 - possible_intervals, 181
 - reset, 179
 - retrieveOnlineData, 179
 - start_date, 181
 - verbose, 181
 - verbose_print, 179
 - writeConfig, 180
 - writeConfigItem, 180
- skdaccess::framework::data_class::DataFetcherBase
 - __init__, 338
 - __str__, 338
 - ap_paramList, 341
 - getConfig, 338
 - getConfigItem, 338
 - getMetadata, 339
 - multirun_enabled, 339
 - output, 339
 - perturb, 340
 - reset, 340
 - verbose, 341
 - verbose_print, 340
 - writeConfig, 341
 - writeConfigItem, 341
- skdaccess::framework::data_class::DataFetcherCache
 - __str__, 344
 - ap_paramList, 349
 - cacheData, 344
 - checkIfDataExists, 344
 - getConfig, 345
 - getConfigItem, 345
 - getDataLocation, 346
 - getHDFStorage, 346
 - getMetadata, 346
 - multirun_enabled, 347
 - output, 347
 - perturb, 347
 - reset, 347
 - setDataLocation, 347
 - verbose, 349
 - verbose_print, 348
 - writeConfig, 348
 - writeConfigItem, 348
- skdaccess::framework::data_class::DataFetcherLocal
 - __str__, 350
 - ap_paramList, 354
 - getConfig, 351
 - getConfigItem, 351
 - getDataLocation, 351
 - getMetadata, 352
 - multirun_enabled, 352
 - output, 352
 - perturb, 352
 - reset, 353
 - setDataLocation, 353
 - verbose, 355
 - verbose_print, 353
 - writeConfig, 354

- writeConfigItem, 354
- skdaccess::framework::data_class::DataFetcherStorage
 - __str__, 356
 - ap_paramList, 361
 - downloadFullDataset, 356
 - getConfig, 357
 - getConfigItem, 357
 - getDataLocation, 357
 - getMetadata, 358
 - multirun_enabled, 358
 - output, 358
 - perturb, 359
 - reset, 359
 - setDataLocation, 359
 - verbose, 361
 - verbose_print, 360
 - writeConfig, 360
 - writeConfigItem, 360
- skdaccess::framework::data_class::DataFetcherStream
 - __str__, 362
 - ap_paramList, 366
 - getConfig, 363
 - getConfigItem, 363
 - getMetadata, 363
 - multirun_enabled, 364
 - output, 364
 - perturb, 364
 - reset, 364
 - retrieveOnlineData, 364
 - verbose, 366
 - verbose_print, 365
 - writeConfig, 365
 - writeConfigItem, 365
- skdaccess::framework::data_class::DataWrapperBase
 - __init__, 372
 - __len__, 373
 - addResult, 373
 - constants, 376
 - data, 376
 - get, 373
 - getIterator, 374
 - getResults, 374
 - getRunID, 374
 - info, 374
 - meta_data, 376
 - reset, 375
 - results, 376
 - run_id, 376
 - update, 375
 - updateMetadata, 375
- skdaccess::framework::data_class::ImageWrapper
 - __len__, 379
 - addResult, 380
 - constants, 383
 - data, 383
 - deleteData, 380
 - get, 380
 - getIterator, 381
 - getResults, 381
 - getRunID, 381
 - info, 381
 - meta_data, 383
 - reset, 382
 - results, 383
 - run_id, 384
 - update, 382
 - updateData, 382
 - updateMetadata, 383
- skdaccess::framework::data_class::SeriesDictionary←
 - Wrapper
 - __len__, 392
 - addResult, 392
 - constants, 395
 - data, 395
 - data_names, 396
 - error_names, 396
 - get, 393
 - getIndices, 393
 - getIterator, 393
 - getLength, 393
 - getResults, 394
 - getRunID, 394
 - info, 394
 - meta_data, 396
 - reset, 394
 - results, 396
 - run_id, 396
 - update, 395
 - updateMetadata, 395
- skdaccess::framework::data_class::SeriesWrapper
 - __init__, 398
 - __len__, 398
 - addResult, 399
 - constants, 402
 - data, 402
 - data_names, 402
 - error_names, 402
 - get, 399
 - getIndices, 399
 - getIterator, 400
 - getLength, 400
 - getResults, 400
 - getRunID, 400
 - info, 401
 - meta_data, 403
 - reset, 401
 - results, 403
 - run_id, 403

- update, [401](#)
- updateMetadata, [402](#)
- skdaccess::framework::data_class::TableWrapper
 - __init__, [411](#)
 - __len__, [412](#)
 - addColumn, [412](#)
 - addResult, [412](#)
 - constants, [417](#)
 - data, [417](#)
 - default_columns, [417](#)
 - default_error_columns, [417](#)
 - get, [413](#)
 - getDefaultColumns, [413](#)
 - getDefaultErrorColumns, [413](#)
 - getIterator, [413](#)
 - getLength, [414](#)
 - getResults, [414](#)
 - getRunID, [414](#)
 - info, [414](#)
 - meta_data, [417](#)
 - removeFrames, [415](#)
 - reset, [415](#)
 - results, [417](#)
 - run_id, [418](#)
 - update, [415](#)
 - updateData, [416](#)
 - updateFrames, [416](#)
 - updateMetadata, [416](#)
- skdaccess::framework::data_class::XArrayWrapper
 - __init__, [419](#)
 - __len__, [419](#)
 - addResult, [419](#)
 - constants, [422](#)
 - data, [422](#)
 - get, [420](#)
 - getIterator, [420](#)
 - getResults, [420](#)
 - getRunID, [420](#)
 - index_list, [422](#)
 - info, [421](#)
 - meta_data, [422](#)
 - reset, [421](#)
 - results, [423](#)
 - run_id, [423](#)
 - update, [421](#)
 - updateMetadata, [422](#)
- skdaccess::framework::param_class::AutoList
 - __call__, [66](#)
 - __getitem__, [66](#)
 - __init__, [66](#)
 - __len__, [67](#)
 - __setitem__, [67](#)
 - __str__, [67](#)
 - getAllOptions, [68](#)
 - perturb, [68](#)
 - reset, [68](#)
 - val, [68](#)
 - val_init, [69](#)
 - val_list, [69](#)
- skdaccess::framework::param_class::AutoListCycle
 - __call__, [71](#)
 - __getitem__, [71](#)
 - __init__, [70](#)
 - __len__, [71](#)
 - __setitem__, [72](#)
 - __str__, [72](#)
 - getAllOptions, [72](#)
 - index, [73](#)
 - list_val_list, [73](#)
 - perturb, [72](#)
 - reset, [73](#)
 - val, [73](#)
 - val_init, [73](#)
 - val_list, [74](#)
- skdaccess::framework::param_class::AutoListPermute
 - __call__, [75](#)
 - __getitem__, [75](#)
 - __len__, [75](#)
 - __setitem__, [76](#)
 - __str__, [76](#)
 - getAllOptions, [76](#)
 - perturb, [77](#)
 - reset, [77](#)
 - val, [77](#)
 - val_init, [77](#)
 - val_list, [78](#)
- skdaccess::framework::param_class::AutoListRemove
 - __call__, [79](#)
 - __getitem__, [79](#)
 - __init__, [79](#)
 - __len__, [80](#)
 - __setitem__, [80](#)
 - __str__, [80](#)
 - getAllOptions, [81](#)
 - n, [82](#)
 - perturb, [81](#)
 - reset, [81](#)
 - val, [81](#)
 - val_init, [82](#)
 - val_list, [82](#)
- skdaccess::framework::param_class::AutoListSubset
 - __call__, [83](#)
 - __getitem__, [83](#)
 - __len__, [84](#)
 - __setitem__, [84](#)
 - __str__, [84](#)
 - getAllOptions, [85](#)
 - perturb, [85](#)

- reset, 85
- val, 85
- val_init, 86
- val_list, 86
- skdaccess::framework::param_class::AutoParam
 - __call__, 88
 - __init__, 87
 - __str__, 88
 - perturb, 88
 - reset, 88
 - val, 89
 - val_init, 89
- skdaccess::framework::param_class::AutoParamList
 - __call__, 90
 - __init__, 90
 - __str__, 90
 - perturb, 91
 - reset, 91
 - val, 91
 - val_init, 91
 - val_list, 91
- skdaccess::framework::param_class::AutoParamListCycle
 - __call__, 93
 - __init__, 93
 - __str__, 93
 - current_index, 94
 - perturb, 93
 - reset, 94
 - val, 94
 - val_init, 94
 - val_list, 94
- skdaccess::framework::param_class::AutoParamMinMax
 - __call__, 96
 - __init__, 95
 - __str__, 96
 - decimals, 97
 - n, 97
 - n_max, 97
 - perturb, 96
 - reset, 97
 - val, 97
 - val_init, 97
 - val_max, 98
 - val_min, 98
- skdaccess::geo::era_interim::cache::data_fetcher::Data←
Fetcher
 - __init__, 105
 - __str__, 106
 - ap_paramList, 111
 - cacheData, 106
 - checkIfDataExists, 107
 - data_names, 112
 - date_list, 112
 - getConfig, 107
 - getConfigItem, 107
 - getDataLocation, 108
 - getHDFStorage, 108
 - getMetadata, 109
 - multirun_enabled, 109
 - output, 109
 - password, 112
 - perturb, 109
 - reset, 110
 - setDataLocation, 110
 - username, 112
 - verbose, 112
 - verbose_print, 110
 - writeConfig, 111
 - writeConfigItem, 111
- skdaccess::geo::gldas::data_fetcher::DataFetcher
 - __init__, 193
 - __str__, 194
 - ap_paramList, 198
 - downloadFullDataset, 194
 - end_date, 198
 - getConfig, 194
 - getConfigItem, 194
 - getDataLocation, 195
 - getMetadata, 195
 - multirun_enabled, 195
 - output, 196
 - perturb, 196
 - resample, 198
 - reset, 196
 - setDataLocation, 196
 - start_date, 198
 - verbose, 198
 - verbose_print, 197
 - writeConfig, 197
 - writeConfigItem, 197
- skdaccess::geo::grace::data_fetcher::DataFetcher
 - __init__, 264
 - __str__, 264
 - ap_paramList, 269
 - downloadFullDataset, 264
 - end_date, 269
 - getConfig, 265
 - getConfigItem, 265
 - getDataLocation, 265
 - getMetadata, 266
 - multirun_enabled, 266
 - output, 266
 - perturb, 267
 - reset, 267
 - setDataLocation, 267
 - start_date, 269
 - verbose, 269
 - verbose_print, 268

- writeConfig, 268
- writeConfigItem, 268
- skdaccess::geo::grace::mascon::cache::data_fetcher::DataFetcher
 - __init__, 320
 - __str__, 321
 - ap_paramList, 326
 - cacheData, 321
 - checkIfDataExists, 322
 - end_date, 327
 - getConfig, 322
 - getConfigItem, 322
 - getDataLocation, 323
 - getHDFStorage, 323
 - getMasconPlacement, 323
 - getMetadata, 324
 - mascon_placement_url, 327
 - mascon_url, 327
 - multirun_enabled, 324
 - output, 324
 - perturb, 324
 - reset, 325
 - scale_factor_url, 327
 - setDataLocation, 325
 - start_date, 327
 - verbose, 327
 - verbose_print, 325
 - writeConfig, 326
 - writeConfigItem, 326
- skdaccess::geo::groundwater::data_fetcher::DataFetcher
 - __init__, 126
 - __str__, 127
 - ap_paramList, 132
 - cutoff, 132
 - downloadFullDataset, 127
 - end_date, 132
 - getConfig, 128
 - getConfigItem, 128
 - getDataLocation, 128
 - getMetadata, 129
 - getStationMetadata, 129
 - multirun_enabled, 129
 - output, 129
 - perturb, 130
 - reset, 130
 - setDataLocation, 130
 - start_date, 132
 - verbose, 132
 - verbose_print, 131
 - writeConfig, 131
 - writeConfigItem, 131
- skdaccess::geo::imsdnh::data_fetcher::DataFetcher
 - __init__, 300
 - __str__, 301
 - ap_paramList, 305
 - coordinate_dict, 305
 - downloadFullDataset, 301
 - end_date, 305
 - getConfig, 301
 - getConfigItem, 302
 - getDataLocation, 302
 - getMetadata, 302
 - multirun_enabled, 303
 - output, 303
 - perturb, 303
 - reset, 303
 - setDataLocation, 303
 - start_date, 305
 - verbose, 305
 - verbose_print, 304
 - writeConfig, 304
 - writeConfigItem, 304
- skdaccess::geo::magnetometer::data_fetcher::DataFetcher
 - __init__, 209
 - __str__, 209
 - ap_paramList, 213
 - channels, 214
 - data_type, 214
 - end_time, 214
 - getConfig, 210
 - getConfigItem, 210
 - getDataMetadata, 210
 - getMetadata, 211
 - interval, 214
 - multirun_enabled, 211
 - output, 211
 - perturb, 211
 - reset, 212
 - retrieveOnlineData, 212
 - start_time, 214
 - verbose, 214
 - verbose_print, 212
 - writeConfig, 213
 - writeConfigItem, 213
- skdaccess::geo::mahali::rinex::data_fetcher::DataFetcher
 - __init__, 184
 - __str__, 185
 - ap_paramList, 191
 - cacheData, 185
 - checkIfDataExists, 186
 - date_range, 191
 - end_date, 191
 - generate_links, 191
 - getConfig, 187
 - getConfigItem, 187
 - getDataLocation, 187
 - getHDFStorage, 188

- getMetadata, 188
- multirun_enabled, 188
- output, 189
- perturb, 189
- reset, 189
- setDataLocation, 189
- start_date, 191
- verbose, 191
- verbose_print, 190
- writeConfig, 190
- writeConfigItem, 190
- skdaccess::geo::mahali::rinex::data_wrapper::Data↔
 - Wrapper
 - __len__, 367
 - addResult, 368
 - constants, 370
 - data, 370
 - get, 368
 - getIterator, 368
 - getResults, 369
 - getRunID, 369
 - info, 369
 - meta_data, 371
 - reset, 369
 - results, 371
 - run_id, 371
 - update, 370
 - updateMetadata, 370
- skdaccess::geo::mahali::tec::data_fetcher::DataFetcher
 - __init__, 201
 - __str__, 202
 - ap_paramList, 207
 - cacheData, 202
 - checkIfDataExists, 202
 - date_range, 207
 - end_date, 207
 - getConfig, 203
 - getConfigItem, 203
 - getDataLocation, 204
 - getHDFStorage, 204
 - getMetadata, 204
 - multirun_enabled, 205
 - output, 205
 - perturb, 205
 - reset, 205
 - setDataLocation, 205
 - start_date, 207
 - verbose, 207
 - verbose_print, 206
 - writeConfig, 206
 - writeConfigItem, 206
- skdaccess::geo::mahali::temperature::data_fetcher::↔
 - DataFetcher
 - __init__, 168
 - __str__, 168
 - ap_paramList, 172
 - end_date, 172
 - getConfig, 168
 - getConfigItem, 168
 - getMetadata, 169
 - multirun_enabled, 169
 - output, 169
 - perturb, 170
 - reset, 170
 - retrieveOnlineData, 170
 - start_date, 172
 - verbose, 172
 - verbose_print, 171
 - writeConfig, 171
 - writeConfigItem, 171
- skdaccess::geo::modis::cache::cloud_mask::data_↔
 - fetcher::DataFetcher
 - __init__, 182
- skdaccess::geo::modis::cache::cloud_opacity::data_↔
 - fetcher::DataFetcher
 - __init__, 173
- skdaccess::geo::modis::cache::data_fetcher::DataFetcher
 - __init__, 233
 - __str__, 234
 - ap_paramList, 240
 - cacheData, 234
 - checkIfDataExists, 235
 - daynightboth, 240
 - end_date, 240
 - find_data, 235
 - getConfig, 236
 - getConfigItem, 236
 - getDataLocation, 237
 - getHDFStorage, 237
 - getMetadata, 237
 - grid, 240
 - grid_fill, 240
 - modis_id, 240
 - modis_identifier, 241
 - modis_platform, 241
 - multirun_enabled, 238
 - output, 238
 - perturb, 238
 - reset, 238
 - setDataLocation, 238
 - start_date, 241
 - use_long_name, 241
 - variable_list, 241
 - verbose, 241
 - verbose_print, 239
 - writeConfig, 239
 - writeConfigItem, 239

skdaccess::geo::modis::cache::reflectance::data_↔
 fetcher::DataFetcher
 __init__, 166
 skdaccess::geo::modis::stream::cloud_mask::data_↔
 fetcher::DataFetcher
 __init__, 174
 skdaccess::geo::modis::stream::cloud_opacity::data_↔
 fetcher::DataFetcher
 __init__, 115
 skdaccess::geo::modis::stream::data_fetcher::Data↔
 Fetcher
 __init__, 313
 __str__, 313
 ap_paramList, 317
 daynightboth, 317
 end_date, 317
 getConfig, 313
 getConfigItem, 314
 getMetadata, 314
 grid, 317
 grid_fill, 317
 modis_id, 318
 modis_identifier, 318
 modis_platform, 318
 multirun_enabled, 314
 output, 315
 perturb, 315
 reset, 315
 retrieveOnlineData, 315
 start_date, 318
 use_long_name, 318
 variable_list, 318
 verbose, 318
 verbose_print, 316
 writeConfig, 316
 writeConfigItem, 316
 skdaccess::geo::modis::stream::reflectance::data_↔
 fetcher::DataFetcher
 __init__, 242
 skdaccess::geo::ngl_gps::data_fetcher::DataFetcher
 __init__, 224
 __str__, 225
 ap_paramList, 230
 data_type, 230
 downloadFullDataset, 225
 end_date, 230
 getAntennaLogs, 225
 getConfig, 226
 getConfigItem, 226
 getDataLocation, 226
 getMetadata, 227
 getStationMetadata, 227
 lat_range, 230
 lon_range, 230
 mdyratio, 230
 multirun_enabled, 227
 output, 228
 perturb, 228
 reset, 228
 setDataLocation, 228
 start_date, 231
 verbose, 231
 verbose_print, 229
 writeConfig, 229
 writeConfigItem, 229
 skdaccess::geo::pbo::data_fetcher::DataFetcher
 __init__, 253
 __str__, 255
 antenna_info, 261
 ap_paramList, 261
 default_columns, 261
 default_error_columns, 261
 downloadFullDataset, 255
 getAntennaLogs, 256
 getConfig, 256
 getConfigItem, 256
 getDataLocation, 257
 getInfo, 257
 getMetadata, 257
 getStationMetadata, 257
 index_date_only, 261
 meta_data, 261
 multirun_enabled, 258
 output, 258
 perturb, 258
 reset, 258
 setDataLocation, 259
 setStationList, 259
 station_list, 261
 use_progress_bar, 262
 verbose, 262
 verbose_print, 259
 writeConfig, 260
 writeConfigItem, 260
 skdaccess::geo::sentinel_1::cache::data_fetcher::Data↔
 Fetcher
 __init__, 329
 __str__, 330
 ap_paramList, 335
 cacheData, 330
 checkIfDataExists, 331
 getConfig, 331
 getConfigItem, 331
 getDataLocation, 332
 getHDFSStorage, 332
 getMetadata, 332
 local_paths, 335
 multirun_enabled, 333

- output, 333
- password, 335
- perturb, 333
- polarization, 336
- reset, 333
- satellite_url_list, 336
- setDataLocation, 334
- swath, 336
- url_list, 336
- username, 336
- verbose, 336
- verbose_print, 334
- writeConfig, 334
- writeConfigItem, 335
- skdaccess::geo::srtm::cache::data_fetcher::DataFetcher
 - __init__, 157
 - __str__, 158
 - ap_paramList, 163
 - arcsecond_sampling, 163
 - cacheData, 158
 - checkIfDataExists, 159
 - getConfig, 159
 - getConfigItem, 159
 - getDataLocation, 160
 - getHDFStorage, 160
 - getMetadata, 160
 - lat_tile_end, 164
 - lat_tile_start, 164
 - lon_tile_end, 164
 - lon_tile_start, 164
 - mask_water, 164
 - multirun_enabled, 161
 - output, 161
 - password, 164
 - perturb, 161
 - reset, 161
 - setDataLocation, 162
 - store_geolocation_grids, 165
 - username, 165
 - verbose, 165
 - verbose_print, 162
 - writeConfig, 162
 - writeConfigItem, 163
- skdaccess::geo::uavsar::cache::data_fetcher::DataFetcher
 - __init__, 216
 - __str__, 217
 - ap_paramList, 222
 - cacheData, 217
 - checkIfDataExists, 217
 - getConfig, 218
 - getConfigItem, 218
 - getDataLocation, 219
 - getHDFStorage, 219
 - getMetadata, 219
 - llh_url, 222
 - memmap, 222
 - metadata_url_list, 222
 - multirun_enabled, 220
 - output, 220
 - perturb, 220
 - reset, 220
 - setDataLocation, 220
 - slc_url_list, 222
 - verbose, 222
 - verbose_print, 221
 - writeConfig, 221
 - writeConfigItem, 221
- skdaccess::geo::wyoming_sounding::cache::data_fetcher::DataFetcher
 - __init__, 244
 - __str__, 245
 - ap_paramList, 250
 - cacheData, 245
 - checkIfDataExists, 246
 - day_end, 250
 - day_start, 250
 - end_hour, 251
 - getConfig, 246
 - getConfigItem, 246
 - getDataLocation, 247
 - getHDFStorage, 247
 - getMetadata, 247
 - month_list, 251
 - multirun_enabled, 248
 - output, 248
 - perturb, 248
 - reset, 248
 - setDataLocation, 249
 - start_hour, 251
 - station_number, 251
 - verbose, 251
 - verbose_print, 249
 - writeConfig, 249
 - writeConfigItem, 250
 - year_list, 251
- skdaccess::geo::wyoming_sounding::stream::data_fetcher::DataFetcher
 - __init__, 150
 - __str__, 150
 - ap_paramList, 154
 - day_end, 154
 - day_start, 154
 - end_hour, 155
 - getConfig, 150
 - getConfigItem, 151
 - getMetadata, 151
 - month_list, 155

- multirun_enabled, 151
- output, 152
- perturb, 152
- reset, 152
- retrieveOnlineData, 153
- start_hour, 155
- station_number, 155
- verbose, 155
- verbose_print, 153
- writeConfig, 153
- writeConfigItem, 154
- year_list, 155
- skdaccess::planetary::ode::cache::data_fetcher::Data↔
 - Fetcher
 - __init__, 277
 - __str__, 277
 - ap_paramList, 284
 - cacheData, 278
 - checkIfDataExists, 278
 - eastern_lon, 284
 - file_name, 284
 - getConfig, 279
 - getConfigItem, 279
 - getDataLocation, 279
 - getHDFStorage, 281
 - getMetadata, 281
 - instrument, 284
 - max_lat, 284
 - max_ob_time, 284
 - min_lat, 285
 - min_ob_time, 285
 - mission, 285
 - multirun_enabled, 281
 - number_product_limit, 285
 - output, 282
 - perturb, 282
 - product_id, 285
 - product_type, 285
 - remove_ndv, 285
 - reset, 282
 - result_offset_number, 286
 - setDataLocation, 282
 - target, 286
 - verbose, 286
 - verbose_print, 283
 - western_lon, 286
 - writeConfig, 283
 - writeConfigItem, 283
- skdaccess::solar::sdo::data_fetcher::DataFetcher
 - __init__, 99
 - __str__, 100
 - ap_paramList, 103
 - getConfig, 100
 - getConfigItem, 100
 - getMetadata, 101
 - multirun_enabled, 101
 - output, 101
 - perturb, 101
 - reset, 102
 - retrieveOnlineData, 102
 - verbose, 104
 - verbose_print, 102
 - writeConfig, 103
 - writeConfigItem, 103
- skdaccess::utilities::file_browser::FileBrowser
 - __init__, 377
 - dirs, 378
 - files, 378
 - path, 378
 - widget, 377
- skdaccess::utilities::file_util
 - openPandasHDFStoreLocking, 31
- skdaccess::utilities::grace_util
 - averageDates, 32
 - computeEWD, 32
 - dateMismatch, 33
 - getStartDate, 33
 - readTellusData, 33
- skdaccess::utilities::gw_util
 - combine_water_heights, 34
- skdaccess::utilities::image_util
 - convertBinCentersToEdges, 36
 - getExtentsFromCentersPlateCarree, 37
 - getGeoTransform, 37
 - lat_spline, 38
 - lon_spline, 38
 - SplineGeolocation, 38
 - x_offset, 38
 - x_spline, 38
 - y_offset, 38
 - y_spline, 38
- skdaccess::utilities::image_util::AffineGlobalCoords
 - __init__, 63
 - getPixelYX, 64
 - getProjectedYX, 64
- skdaccess::utilities::image_util::LinearGeolocation
 - __init__, 388
 - flip_y, 389
 - getExtents, 388
 - getLatLon, 388
 - getYX, 389
 - lat_extents, 389
 - lat_pixel_size, 389
 - len_x, 390
 - len_y, 390
 - lon_extents, 390
 - lon_pixel_size, 390
 - start_lat, 390

- start_lon, 390
- x_offset, 390
- y_offset, 391
- skdaccess::utilities::image_util::SplineLatLon
 - __call__, 408
 - __init__, 407
 - lat_func, 409
 - lon_func, 409
 - x_offset, 409
 - y_offset, 409
- skdaccess::utilities::kepler_util
 - normalize, 39
- skdaccess::utilities::mahali_util
 - convert_date, 40
 - parselonoFile, 40
- skdaccess::utilities::modis_util
 - calibrateModis, 41
 - checkBit, 41
 - createGrid, 42
 - getFileIDs, 42
 - getFileURLs, 43
 - getImageType, 43
 - getModisData, 44
 - readMODISData, 44
 - rescale, 45
- skdaccess::utilities::modis_util::LatLon
 - __call__, 385
 - __init__, 385
 - alat, 386
 - alon, 386
 - lat_data, 386
 - lon_data, 386
 - x_offset, 386
 - y_offset, 386
- skdaccess::utilities::ode_util
 - correct_CRISM_label, 46
 - correct_file_name_case_in_label, 46
 - correct_label_file, 46
 - get_files_urls, 46
 - get_query_url, 47
 - get_raster_array, 47
 - get_raster_extent, 47
 - query_files_urls, 48
 - query_yes_no, 49
- skdaccess::utilities::pbo_util
 - getLatLonRange, 49
 - getROIstations, 50
 - getStationCoords, 50
 - nostab_sys, 51
 - propagateErrors, 52
 - removeAntennaOffset, 52
 - stab_sys, 53
- skdaccess::utilities::sentinel_1_util
 - parseSatelliteData, 53
- skdaccess::utilities::sounding_util
 - generateQueries, 54
- skdaccess::utilities::sounding_util::SoundingParser
 - __init__, 404
 - data_dict, 405
 - handle_data, 404
 - handle_endtag, 405
 - handle_starttag, 405
 - in_header, 405
 - in_pre_tag, 406
 - label, 406
 - metadata_dict, 406
 - read_data, 406
 - tmp, 406
- skdaccess::utilities::srtm_util
 - getSRTMData, 55
 - getSRTMLatLon, 56
 - merge_srtm_tiles, 56
- skdaccess::utilities::support
 - convertToStr, 57
 - join_string, 57
 - progress_bar, 57
 - retrieveCommonDatesHDF, 58
- skdaccess::utilities::tess_utils
 - parseTessData, 58
- skdaccess::utilities::uavsar_util
 - readUAVSARMetadata, 59
- slc_url_list
 - skdaccess::geo::uavsar::cache::data_fetcher::Data←
Fetcher, 222
- solar/sdo/data_fetcher.py, 437
- spacecraft_list
 - skdaccess::astro::voyager::data_fetcher::Data←
Fetcher, 296
- SplineGeolocation
 - skdaccess::utilities::image_util, 38
- stab_sys
 - skdaccess::utilities::pbo_util, 53
- start_date
 - skdaccess::finance::timeseries::stream::Data←
Fetcher, 181
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
198
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
269
 - skdaccess::geo::grace::mascon::cache::data_←
fetcher::DataFetcher, 327
 - skdaccess::geo::groundwater::data_fetcher::Data←
Fetcher, 132
 - skdaccess::geo::imsdnhs::data_fetcher::Data←
Fetcher, 305
 - skdaccess::geo::mahali::rinex::data_fetcher::Data←
Fetcher, 191

- skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 207
- skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 172
- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 241
- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 318
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
231
- start_hour
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 251
 - skdaccess::geo::wyoming_sounding::stream::data_↔
_fetcher::DataFetcher, 155
- start_lat
 - skdaccess::utilities::image_util::LinearGeolocation,
390
- start_lon
 - skdaccess::utilities::image_util::LinearGeolocation,
390
- start_time
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 214
- start_url
 - skdaccess::astro::tess::data::cache::DataFetcher,
298
 - skdaccess::astro::tess::simulated::cache::Data↔
Fetcher, 114
- station_list
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 261
- station_number
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 251
 - skdaccess::geo::wyoming_sounding::stream::data_↔
_fetcher::DataFetcher, 155
- store_geolocation_grids
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 165
- swath
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 336
- target
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 286
- terminal_groundwater_example, 59
 - color, 60
 - data_1, 60
 - data_2, 60
 - dataIt, 60
 - fullIDF, 60
 - fullIDW, 60
 - label_1, 61
 - label_2, 61
 - meta_data, 61
- tmp
 - skdaccess::utilities::sounding_util::SoundingParser,
406
- toi_information
 - skdaccess::astro::tess::generic::cache::DataFetcher,
124
- update
 - skdaccess::framework::data_class::DataWrapper↔
Base, 375
 - skdaccess::framework::data_class::ImageWrapper,
382
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 395
 - skdaccess::framework::data_class::SeriesWrapper,
401
 - skdaccess::framework::data_class::TableWrapper,
415
 - skdaccess::framework::data_class::XArrayWrapper,
421
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 370
- updateData
 - skdaccess::framework::data_class::ImageWrapper,
382
 - skdaccess::framework::data_class::TableWrapper,
416
- updateFrames
 - skdaccess::framework::data_class::TableWrapper,
416
- updateMetadata
 - skdaccess::framework::data_class::DataWrapper↔
Base, 375
 - skdaccess::framework::data_class::ImageWrapper,
383
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 395
 - skdaccess::framework::data_class::SeriesWrapper,
402
 - skdaccess::framework::data_class::TableWrapper,
416
 - skdaccess::framework::data_class::XArrayWrapper,
422
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 370
- url_list
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 336
- use_long_name
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 241

- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 318
- use_progress_bar
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 262
- username
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 112
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 336
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 165
- utilities/file_browser.py, 437
- utilities/file_util.py, 438
- utilities/grace_util.py, 438
- utilities/gw_util.py, 438
- utilities/image_util.py, 439
- utilities/kepler_util.py, 439
- utilities/mahali_util.py, 440
- utilities/modis_util.py, 440
- utilities/ode_util.py, 441
- utilities/pbo_util.py, 441
- utilities/sentinel_1_util.py, 442
- utilities/sounding_util.py, 442
- utilities/srtm_util.py, 442
- utilities/support.py, 443
- utilities/tess_utils.py, 443
- utilities/uavsar_util.py, 443
- val
 - skdaccess::framework::param_class::AutoList, 68
 - skdaccess::framework::param_class::AutoListCycle,
73
 - skdaccess::framework::param_class::AutoList↔
Permute, 77
 - skdaccess::framework::param_class::AutoList↔
Remove, 81
 - skdaccess::framework::param_class::AutoList↔
Subset, 85
 - skdaccess::framework::param_class::AutoParam, 89
 - skdaccess::framework::param_class::AutoParamList,
91
 - skdaccess::framework::param_class::AutoParam↔
ListCycle, 94
 - skdaccess::framework::param_class::AutoParam↔
MinMax, 97
- val_init
 - skdaccess::framework::param_class::AutoList, 69
 - skdaccess::framework::param_class::AutoListCycle,
73
 - skdaccess::framework::param_class::AutoList↔
Permute, 77
 - skdaccess::framework::param_class::AutoList↔
Remove, 82
- skdaccess::framework::param_class::AutoList↔
Subset, 86
- skdaccess::framework::param_class::AutoParam, 89
- skdaccess::framework::param_class::AutoParamList,
91
- skdaccess::framework::param_class::AutoParam↔
ListCycle, 94
- val_max
 - skdaccess::framework::param_class::AutoParam↔
MinMax, 98
- val_min
 - skdaccess::framework::param_class::AutoParam↔
MinMax, 98
- variable_list
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 241
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 318
- verbose
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
141
 - skdaccess::astro::spectra::stream::DataFetcher, 311
 - skdaccess::astro::tess::generic::cache::DataFetcher,
124
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 296
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, 148
 - skdaccess::engineering::webcam::mit_sailing↔
::stream::DataFetcher, 275
 - skdaccess::finance::timeseries::stream::Data↔
Fetcher, 181
 - skdaccess::framework::data_class::DataFetcher↔
Base, 341
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 349
 - skdaccess::framework::data_class::DataFetcher↔
Local, 355

- skdaccess::framework::data_class::DataFetcher↔
Storage, 361
- skdaccess::framework::data_class::DataFetcher↔
Stream, 366
- skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 112
- skdaccess::geo::gldas::data_fetcher::DataFetcher,
198
- skdaccess::geo::grace::data_fetcher::DataFetcher,
269
- skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 327
- skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 132
- skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 305
- skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 214
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 191
- skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 207
- skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 172
- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 241
- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 318
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
231
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 262
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 336
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 165
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 222
- skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 251
- skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, 155
- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 286
- skdaccess::solar::sdo::data_fetcher::DataFetcher,
104
- verbose_print
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
140
 - skdaccess::astro::spectra::stream::DataFetcher, 310
 - skdaccess::astro::tess::generic::cache::DataFetcher,
123
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 294
 - skdaccess::engineering::la::generic::stream::Data↔
Fetcher, 146
- skdaccess::engineering::webcam::mit_sailing↔
::stream::DataFetcher, 273
- skdaccess::finance::timeseries::stream::Data↔
Fetcher, 179
- skdaccess::framework::data_class::DataFetcher↔
Base, 340
- skdaccess::framework::data_class::DataFetcher↔
Cache, 348
- skdaccess::framework::data_class::DataFetcher↔
Local, 353
- skdaccess::framework::data_class::DataFetcher↔
Storage, 360
- skdaccess::framework::data_class::DataFetcher↔
Stream, 365
- skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 110
- skdaccess::geo::gldas::data_fetcher::DataFetcher,
197
- skdaccess::geo::grace::data_fetcher::DataFetcher,
268
- skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 325
- skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 131
- skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 304
- skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 212
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 190
- skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 206
- skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 171
- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 239
- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 316
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
229
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 259
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 334
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 162
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 221
- skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 249
- skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, 153
- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 283

- skdaccess::solar::sdo::data_fetcher::DataFetcher, 102
- western_lon
 - skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 286
- widget
 - skdaccess::utilities::file_browser::FileBrowser, 377
- writeConfig
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 140
 - skdaccess::astro::spectra::stream::DataFetcher, 310
 - skdaccess::astro::tess::generic::cache::DataFetcher, 123
 - skdaccess::astro::voyager::data_fetcher::DataFetcher, 295
 - skdaccess::engineering::la::generic::stream::DataFetcher, 146
 - skdaccess::engineering::webcam::mit_sailing::stream::DataFetcher, 274
 - skdaccess::finance::timeseries::stream::DataFetcher, 180
 - skdaccess::framework::data_class::DataFetcherBase, 341
 - skdaccess::framework::data_class::DataFetcherCache, 348
 - skdaccess::framework::data_class::DataFetcherLocal, 354
 - skdaccess::framework::data_class::DataFetcherStorage, 360
 - skdaccess::framework::data_class::DataFetcherStream, 365
 - skdaccess::geo::era_interim::cache::data_fetcher::DataFetcher, 111
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 197
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 268
 - skdaccess::geo::grace::mascon::cache::data_fetcher::DataFetcher, 326
 - skdaccess::geo::groundwater::data_fetcher::DataFetcher, 131
 - skdaccess::geo::imsdnhs::data_fetcher::DataFetcher, 304
 - skdaccess::geo::magnetometer::data_fetcher::DataFetcher, 213
 - skdaccess::geo::mahali::rinex::data_fetcher::DataFetcher, 190
 - skdaccess::geo::mahali::tec::data_fetcher::DataFetcher, 206
 - skdaccess::geo::mahali::temperature::data_fetcher::DataFetcher, 171
 - skdaccess::geo::modis::cache::data_fetcher::DataFetcher, 239
 - skdaccess::geo::modis::stream::data_fetcher::DataFetcher, 316
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 229
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 260
 - skdaccess::geo::sentinel_1::cache::data_fetcher::DataFetcher, 334
 - skdaccess::geo::srtm::cache::data_fetcher::DataFetcher, 162
 - skdaccess::geo::uavsar::cache::data_fetcher::DataFetcher, 221
 - skdaccess::geo::wyoming_sounding::cache::data_fetcher::DataFetcher, 249
 - skdaccess::geo::wyoming_sounding::stream::data_fetcher::DataFetcher, 153
 - skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 283
 - skdaccess::solar::sdo::data_fetcher::DataFetcher, 103
- writeConfigItem
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 140
 - skdaccess::astro::spectra::stream::DataFetcher, 310
 - skdaccess::astro::tess::generic::cache::DataFetcher, 124
 - skdaccess::astro::voyager::data_fetcher::DataFetcher, 295
 - skdaccess::engineering::la::generic::stream::DataFetcher, 146
 - skdaccess::engineering::webcam::mit_sailing::stream::DataFetcher, 274
 - skdaccess::finance::timeseries::stream::DataFetcher, 180
 - skdaccess::framework::data_class::DataFetcherBase, 341
 - skdaccess::framework::data_class::DataFetcherCache, 348
 - skdaccess::framework::data_class::DataFetcherLocal, 354
 - skdaccess::framework::data_class::DataFetcherStorage, 360
 - skdaccess::framework::data_class::DataFetcherStream, 365
 - skdaccess::geo::era_interim::cache::data_fetcher::DataFetcher, 111
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 197
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 268
 - skdaccess::geo::grace::mascon::cache::data_fetcher::DataFetcher, 326
 - skdaccess::geo::groundwater::data_fetcher::DataFetcher, 131
 - skdaccess::geo::imsdnhs::data_fetcher::DataFetcher, 304

- Fetcher, [304](#)
- skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, [213](#)
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, [190](#)
- skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, [206](#)
- skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, [171](#)
- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [239](#)
- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, [316](#)
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
[229](#)
- skdaccess::geo::pbo::data_fetcher::DataFetcher, [260](#)
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [335](#)
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [163](#)
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, [221](#)
- skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, [250](#)
- skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, [154](#)
- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [283](#)
- skdaccess::solar::sdo::data_fetcher::DataFetcher,
[103](#)
- x_offset
 - skdaccess::utilities::image_util, [38](#)
 - skdaccess::utilities::image_util::LinearGeolocation,
[390](#)
 - skdaccess::utilities::image_util::SplineLatLon, [409](#)
 - skdaccess::utilities::modis_util::LatLon, [386](#)
- x_spline
 - skdaccess::utilities::image_util, [38](#)
- y_offset
 - skdaccess::utilities::image_util, [38](#)
 - skdaccess::utilities::image_util::LinearGeolocation,
[391](#)
 - skdaccess::utilities::image_util::SplineLatLon, [409](#)
 - skdaccess::utilities::modis_util::LatLon, [386](#)
- y_spline
 - skdaccess::utilities::image_util, [38](#)
- year_list
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [296](#)
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, [251](#)
 - skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, [155](#)