

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	13
5.5	skdaccess.astro.voyager Namespace Reference	14
5.6	skdaccess.astro.voyager.data_fetcher Namespace Reference	14
5.7	skdaccess.framework Namespace Reference	14
5.8	skdaccess.framework.data_class Namespace Reference	14
5.9	skdaccess.framework.param_class Namespace Reference	15
5.10	skdaccess.geo Namespace Reference	15

5.11	skdaccess.geo.era_interim Namespace Reference	15
5.12	skdaccess.geo.era_interim.cache Namespace Reference	16
5.13	skdaccess.geo.era_interim.cache.data_fetcher Namespace Reference	16
5.14	skdaccess.geo.gldas Namespace Reference	16
5.15	skdaccess.geo.gldas.data_fetcher Namespace Reference	16
5.16	skdaccess.geo.grace Namespace Reference	16
5.17	skdaccess.geo.grace.data_fetcher Namespace Reference	16
5.18	skdaccess.geo.grace.mascon Namespace Reference	17
5.19	skdaccess.geo.grace.mascon.cache Namespace Reference	17
5.20	skdaccess.geo.grace.mascon.cache.data_fetcher Namespace Reference	17
5.21	skdaccess.geo.groundwater Namespace Reference	17
5.22	skdaccess.geo.groundwater.data_fetcher Namespace Reference	17
5.23	skdaccess.geo.imsdnhs Namespace Reference	17
5.24	skdaccess.geo.imsdnhs.data_fetcher Namespace Reference	18
5.25	skdaccess.geo.magnetometer Namespace Reference	18
5.26	skdaccess.geo.magnetometer.data_fetcher Namespace Reference	18
5.27	skdaccess.geo.mahali Namespace Reference	18
5.28	skdaccess.geo.mahali.rinex Namespace Reference	18
5.29	skdaccess.geo.mahali.rinex.data_fetcher Namespace Reference	19
5.30	skdaccess.geo.mahali.rinex.data_wrapper Namespace Reference	19
5.31	skdaccess.geo.mahali.tec Namespace Reference	19
5.32	skdaccess.geo.mahali.tec.data_fetcher Namespace Reference	19
5.33	skdaccess.geo.mahali.temperature Namespace Reference	19
5.34	skdaccess.geo.mahali.temperature.data_fetcher Namespace Reference	19
5.35	skdaccess.geo.modis Namespace Reference	20
5.36	skdaccess.geo.modis.cache Namespace Reference	20
5.37	skdaccess.geo.modis.cache.cloud_mask Namespace Reference	20
5.38	skdaccess.geo.modis.cache.cloud_mask.data_fetcher Namespace Reference	20

5.39 skdaccess.geo.modis.cache.cloud_opacity Namespace Reference	20
5.40 skdaccess.geo.modis.cache.cloud_opacity.data_fetcher Namespace Reference	20
5.41 skdaccess.geo.modis.cache.data_fetcher Namespace Reference	21
5.42 skdaccess.geo.modis.cache.reflectance Namespace Reference	21
5.43 skdaccess.geo.modis.cache.reflectance.data_fetcher Namespace Reference	21
5.44 skdaccess.geo.modis.stream Namespace Reference	21
5.45 skdaccess.geo.modis.stream.cloud_mask Namespace Reference	21
5.46 skdaccess.geo.modis.stream.cloud_mask.data_fetcher Namespace Reference	21
5.47 skdaccess.geo.modis.stream.cloud_opacity Namespace Reference	22
5.48 skdaccess.geo.modis.stream.cloud_opacity.data_fetcher Namespace Reference	22
5.49 skdaccess.geo.modis.stream.data_fetcher Namespace Reference	22
5.50 skdaccess.geo.modis.stream.reflectance Namespace Reference	22
5.51 skdaccess.geo.modis.stream.reflectance.data_fetcher Namespace Reference	22
5.52 skdaccess.geo.ngl_gps Namespace Reference	22
5.53 skdaccess.geo.ngl_gps.data_fetcher Namespace Reference	23
5.54 skdaccess.geo.pbo Namespace Reference	23
5.55 skdaccess.geo.pbo.data_fetcher Namespace Reference	23
5.56 skdaccess.geo.sentinel_1 Namespace Reference	23
5.57 skdaccess.geo.sentinel_1.cache Namespace Reference	23
5.58 skdaccess.geo.sentinel_1.cache.data_fetcher Namespace Reference	23
5.59 skdaccess.geo.srtm Namespace Reference	24
5.60 skdaccess.geo.srtm.cache Namespace Reference	24
5.61 skdaccess.geo.srtm.cache.data_fetcher Namespace Reference	24
5.62 skdaccess.geo.uavsar Namespace Reference	24
5.63 skdaccess.geo.uavsar.cache Namespace Reference	24
5.64 skdaccess.geo.uavsar.cache.data_fetcher Namespace Reference	24
5.65 skdaccess.geo.wyoming_sounding Namespace Reference	25
5.66 skdaccess.geo.wyoming_sounding.cache Namespace Reference	25

5.67	skdaccess.geo.wyoming_sounding.cache.data_fetcher Namespace Reference	25
5.68	skdaccess.geo.wyoming_sounding.stream Namespace Reference	25
5.69	skdaccess.geo.wyoming_sounding.stream.data_fetcher Namespace Reference	25
5.70	skdaccess.planetary Namespace Reference	25
5.71	skdaccess.planetary.ode Namespace Reference	26
5.72	skdaccess.planetary.ode.cache Namespace Reference	26
5.73	skdaccess.planetary.ode.cache.data_fetcher Namespace Reference	26
5.74	skdaccess.solar Namespace Reference	26
5.75	skdaccess.solar.sdo Namespace Reference	26
5.76	skdaccess.solar.sdo.data_fetcher Namespace Reference	26
5.77	skdaccess.utilities Namespace Reference	27
5.78	skdaccess.utilities.file_browser Namespace Reference	27
5.79	skdaccess.utilities.grace_util Namespace Reference	27
5.79.1	Function Documentation	27
5.79.1.1	averageDates()	28
5.79.1.2	computeEWD()	28
5.79.1.3	dateMismatch()	28
5.79.1.4	getStartEndDate()	30
5.79.1.5	readTellusData()	30
5.80	skdaccess.utilities.gw_util Namespace Reference	31
5.80.1	Function Documentation	31
5.80.1.1	combine_water_heights()	31
5.81	skdaccess.utilities.image_util Namespace Reference	31
5.81.1	Function Documentation	32
5.81.1.1	convertBinCentersToEdges()	32
5.81.1.2	getExtentsFromCentersPlateCarree()	33
5.81.1.3	getGeoTransform()	33
5.81.1.4	SplineGeolocation()	33

5.81.2	Variable Documentation	34
5.81.2.1	lat_spline	34
5.81.2.2	lon_spline	34
5.81.2.3	x_offset	34
5.81.2.4	x_spline	34
5.81.2.5	y_offset	34
5.81.2.6	y_spline	34
5.82	skdaccess.utilities.kepler_util Namespace Reference	35
5.82.1	Function Documentation	35
5.82.1.1	normalize()	35
5.83	skdaccess.utilities.mahali_util Namespace Reference	35
5.83.1	Function Documentation	35
5.83.1.1	convert_date()	35
5.83.1.2	parselonoFile()	36
5.84	skdaccess.utilities.modis_util Namespace Reference	36
5.84.1	Function Documentation	36
5.84.1.1	calibrateModis()	37
5.84.1.2	checkBit()	37
5.84.1.3	createGrid()	37
5.84.1.4	getFileIDs()	38
5.84.1.5	getFileURLs()	39
5.84.1.6	getImageType()	39
5.84.1.7	getModisData()	40
5.84.1.8	readMODISData()	40
5.84.1.9	rescale()	41
5.85	skdaccess.utilities.ode_util Namespace Reference	41
5.85.1	Function Documentation	41
5.85.1.1	correct_CRISM_label()	42

5.85.1.2	correct_file_name_case_in_label()	42
5.85.1.3	correct_label_file()	42
5.85.1.4	get_files_urls()	42
5.85.1.5	get_query_url()	43
5.85.1.6	get_raster_array()	43
5.85.1.7	get_raster_extent()	43
5.85.1.8	query_files_urls()	44
5.85.1.9	query_yes_no()	45
5.86	skdaccess.utilities.pbo_util Namespace Reference	45
5.86.1	Function Documentation	45
5.86.1.1	getLatLonRange()	45
5.86.1.2	getROIstations()	46
5.86.1.3	getStationCoords()	46
5.86.1.4	nostab_sys()	47
5.86.1.5	propagateErrors()	47
5.86.1.6	removeAntennaOffset()	48
5.86.1.7	stab_sys()	48
5.87	skdaccess.utilities.sentinel_1_util Namespace Reference	49
5.87.1	Function Documentation	49
5.87.1.1	parseSatelliteData()	49
5.88	skdaccess.utilities.sounding_util Namespace Reference	50
5.88.1	Function Documentation	50
5.88.1.1	generateQueries()	50
5.89	skdaccess.utilities.srtm_util Namespace Reference	51
5.89.1	Function Documentation	51
5.89.1.1	getSRTMData()	51
5.89.1.2	getSRTMLatLon()	52
5.89.1.3	merge_srtm_tiles()	52
5.90	skdaccess.utilities.support Namespace Reference	52
5.90.1	Function Documentation	52
5.90.1.1	convertToStr()	53
5.90.1.2	progress_bar()	53
5.90.1.3	retrieveCommonDatesHDF()	53
5.91	skdaccess.utilities.uavsar_util Namespace Reference	53
5.91.1	Function Documentation	54
5.91.1.1	readUAVSARMetadata()	54

6	Class Documentation	55
6.1	skdaccess.utilities.image_util.AffineGlobalCoords Class Reference	55
6.1.1	Detailed Description	55
6.1.2	Constructor & Destructor Documentation	55
6.1.2.1	__init__()	56
6.1.3	Member Function Documentation	56
6.1.3.1	getPixelYX()	56
6.1.3.2	getProjectedYX()	56
6.2	skdaccess.framework.param_class.AutoList Class Reference	57
6.2.1	Detailed Description	58
6.2.2	Constructor & Destructor Documentation	58
6.2.2.1	__init__()	58
6.2.3	Member Function Documentation	58
6.2.3.1	__call__()	58
6.2.3.2	__getitem__()	59
6.2.3.3	__len__()	59
6.2.3.4	__setitem__()	59
6.2.3.5	__str__()	60
6.2.3.6	getAllOptions()	60
6.2.3.7	perturb()	60
6.2.3.8	reset()	60
6.2.3.9	val()	61
6.2.4	Member Data Documentation	61
6.2.4.1	val_init	61
6.2.4.2	val_list	61
6.3	skdaccess.framework.param_class.AutoListCycle Class Reference	61
6.3.1	Detailed Description	62
6.3.2	Constructor & Destructor Documentation	62

6.3.2.1	<code>__init__()</code>	62
6.3.3	Member Function Documentation	63
6.3.3.1	<code>__call__()</code>	63
6.3.3.2	<code>__getitem__()</code>	63
6.3.3.3	<code>__len__()</code>	63
6.3.3.4	<code>__setitem__()</code>	64
6.3.3.5	<code>__str__()</code>	64
6.3.3.6	<code>getAllOptions()</code>	64
6.3.3.7	<code>perturb()</code>	65
6.3.3.8	<code>reset()</code>	65
6.3.3.9	<code>val()</code>	65
6.3.4	Member Data Documentation	65
6.3.4.1	<code>index</code>	65
6.3.4.2	<code>list_val_list</code>	65
6.3.4.3	<code>val_init</code>	66
6.3.4.4	<code>val_list</code>	66
6.4	<code>skdaccess.framework.param_class.AutoListPermute</code> Class Reference	66
6.4.1	Detailed Description	67
6.4.2	Member Function Documentation	67
6.4.2.1	<code>__call__()</code>	67
6.4.2.2	<code>__getitem__()</code>	67
6.4.2.3	<code>__len__()</code>	68
6.4.2.4	<code>__setitem__()</code>	68
6.4.2.5	<code>__str__()</code>	68
6.4.2.6	<code>getAllOptions()</code>	69
6.4.2.7	<code>perturb()</code>	69
6.4.2.8	<code>reset()</code>	69
6.4.2.9	<code>val()</code>	69

6.4.3	Member Data Documentation	69
6.4.3.1	val_init	70
6.4.3.2	val_list	70
6.5	skdaccess.framework.param_class.AutoListRemove Class Reference	70
6.5.1	Detailed Description	71
6.5.2	Constructor & Destructor Documentation	71
6.5.2.1	__init__()	71
6.5.3	Member Function Documentation	71
6.5.3.1	__call__()	71
6.5.3.2	__getitem__()	72
6.5.3.3	__len__()	72
6.5.3.4	__setitem__()	72
6.5.3.5	__str__()	73
6.5.3.6	getAllOptions()	73
6.5.3.7	perturb()	73
6.5.3.8	reset()	73
6.5.3.9	val()	74
6.5.4	Member Data Documentation	74
6.5.4.1	n	74
6.5.4.2	val_init	74
6.5.4.3	val_list	74
6.6	skdaccess.framework.param_class.AutoListSubset Class Reference	74
6.6.1	Detailed Description	75
6.6.2	Member Function Documentation	75
6.6.2.1	__call__()	75
6.6.2.2	__getitem__()	76
6.6.2.3	__len__()	76
6.6.2.4	__setitem__()	76

6.6.2.5	<code>__str__()</code>	77
6.6.2.6	<code>getAllOptions()</code>	77
6.6.2.7	<code>perturb()</code>	77
6.6.2.8	<code>reset()</code>	77
6.6.2.9	<code>val()</code>	78
6.6.3	Member Data Documentation	78
6.6.3.1	<code>val_init</code>	78
6.6.3.2	<code>val_list</code>	78
6.7	<code>skdaccess.framework.param_class.AutoParam</code> Class Reference	78
6.7.1	Detailed Description	79
6.7.2	Constructor & Destructor Documentation	79
6.7.2.1	<code>__init__()</code>	79
6.7.3	Member Function Documentation	80
6.7.3.1	<code>__call__()</code>	80
6.7.3.2	<code>__str__()</code>	80
6.7.3.3	<code>perturb()</code>	80
6.7.3.4	<code>reset()</code>	80
6.7.4	Member Data Documentation	81
6.7.4.1	<code>val</code>	81
6.7.4.2	<code>val_init</code>	81
6.8	<code>skdaccess.framework.param_class.AutoParamList</code> Class Reference	81
6.8.1	Detailed Description	82
6.8.2	Constructor & Destructor Documentation	82
6.8.2.1	<code>__init__()</code>	82
6.8.3	Member Function Documentation	82
6.8.3.1	<code>__call__()</code>	82
6.8.3.2	<code>__str__()</code>	83
6.8.3.3	<code>perturb()</code>	83

6.8.3.4	reset()	83
6.8.4	Member Data Documentation	83
6.8.4.1	val	83
6.8.4.2	val_init	83
6.8.4.3	val_list	84
6.9	skdaccess.framework.param_class.AutoParamListCycle Class Reference	84
6.9.1	Detailed Description	84
6.9.2	Constructor & Destructor Documentation	85
6.9.2.1	__init__()	85
6.9.3	Member Function Documentation	85
6.9.3.1	__call__()	85
6.9.3.2	__str__()	85
6.9.3.3	perturb()	86
6.9.3.4	reset()	86
6.9.4	Member Data Documentation	86
6.9.4.1	current_index	86
6.9.4.2	val	86
6.9.4.3	val_init	86
6.9.4.4	val_list	86
6.10	skdaccess.framework.param_class.AutoParamMinMax Class Reference	87
6.10.1	Detailed Description	87
6.10.2	Constructor & Destructor Documentation	87
6.10.2.1	__init__()	88
6.10.3	Member Function Documentation	88
6.10.3.1	__call__()	88
6.10.3.2	__str__()	88
6.10.3.3	perturb()	89
6.10.3.4	reset()	89

6.10.4	Member Data Documentation	89
6.10.4.1	decimals	89
6.10.4.2	n	89
6.10.4.3	n_max	89
6.10.4.4	val	89
6.10.4.5	val_init	90
6.10.4.6	val_max	90
6.10.4.7	val_min	90
6.11	skdaccess.geo.gldas.DataFetcher Class Reference	90
6.11.1	Detailed Description	91
6.11.2	Constructor & Destructor Documentation	91
6.11.2.1	__init__()	92
6.11.3	Member Function Documentation	92
6.11.3.1	__str__()	92
6.11.3.2	downloadFullDataset()	92
6.11.3.3	getConfig()	93
6.11.3.4	getDataLocation()	93
6.11.3.5	getMetadata()	93
6.11.3.6	multirun_enabled()	94
6.11.3.7	output()	94
6.11.3.8	perturb()	94
6.11.3.9	reset()	94
6.11.3.10	setDataLocation()	94
6.11.3.11	verbose_print()	95
6.11.3.12	writeConfig()	95
6.11.4	Member Data Documentation	95
6.11.4.1	ap_paramList	95
6.11.4.2	end_date	96

6.11.4.3	resample	96
6.11.4.4	start_date	96
6.11.4.5	verbose	96
6.12	skdaccess.geo.sentinel_1.cache.DataFetcher Class Reference	96
6.12.1	Detailed Description	98
6.12.2	Constructor & Destructor Documentation	98
6.12.2.1	__init__()	98
6.12.3	Member Function Documentation	98
6.12.3.1	__str__()	98
6.12.3.2	cacheData()	99
6.12.3.3	checkIfDataExists()	99
6.12.3.4	getConfig()	100
6.12.3.5	getDataLocation()	100
6.12.3.6	getHDFStorage()	100
6.12.3.7	getMetadata()	101
6.12.3.8	multirun_enabled()	101
6.12.3.9	output()	101
6.12.3.10	perturb()	102
6.12.3.11	reset()	102
6.12.3.12	setDataLocation()	102
6.12.3.13	verbose_print()	102
6.12.3.14	writeConfig()	103
6.12.4	Member Data Documentation	103
6.12.4.1	ap_paramList	103
6.12.4.2	local_paths	103
6.12.4.3	password	103
6.12.4.4	polarization	104
6.12.4.5	satellite_url_list	104

6.12.4.6	swath	104
6.12.4.7	url_list	104
6.12.4.8	username	104
6.12.4.9	verbose	104
6.13	skdaccess.geo.groundwater.DataFetcher Class Reference	105
6.13.1	Detailed Description	106
6.13.2	Constructor & Destructor Documentation	106
6.13.2.1	__init__()	106
6.13.3	Member Function Documentation	106
6.13.3.1	__str__()	107
6.13.3.2	downloadFullDataset()	107
6.13.3.3	getConfig()	107
6.13.3.4	getDataLocation()	107
6.13.3.5	getMetadata()	108
6.13.3.6	getStationMetadata()	108
6.13.3.7	multirun_enabled()	108
6.13.3.8	output()	109
6.13.3.9	perturb()	109
6.13.3.10	reset()	109
6.13.3.11	setDataLocation()	109
6.13.3.12	verbose_print()	110
6.13.3.13	writeConfig()	110
6.13.4	Member Data Documentation	110
6.13.4.1	ap_paramList	110
6.13.4.2	cutoff	110
6.13.4.3	end_date	111
6.13.4.4	start_date	111
6.13.4.5	verbose	111

6.14	skdaccess.geo.srtm.cache.DataFetcher Class Reference	111
6.14.1	Detailed Description	113
6.14.2	Constructor & Destructor Documentation	113
6.14.2.1	__init__()	113
6.14.3	Member Function Documentation	113
6.14.3.1	__str__()	113
6.14.3.2	cacheData()	114
6.14.3.3	checkIfDataExists()	114
6.14.3.4	getConfig()	115
6.14.3.5	getDataLocation()	115
6.14.3.6	getHDFSStorage()	115
6.14.3.7	getMetadata()	116
6.14.3.8	multirun_enabled()	116
6.14.3.9	output()	116
6.14.3.10	perturb()	117
6.14.3.11	reset()	117
6.14.3.12	setDataLocation()	117
6.14.3.13	verbose_print()	117
6.14.3.14	writeConfig()	118
6.14.4	Member Data Documentation	118
6.14.4.1	ap_paramList	118
6.14.4.2	arcsecond_sampling	118
6.14.4.3	lat_tile_end	119
6.14.4.4	lat_tile_start	119
6.14.4.5	lon_tile_end	119
6.14.4.6	lon_tile_start	119
6.14.4.7	mask_water	119
6.14.4.8	password	119

6.14.4.9	store_geolocation_grids	119
6.14.4.10	username	120
6.14.4.11	verbose	120
6.15	skdaccess.geo.uavsar.cache.DataFetcher Class Reference	120
6.15.1	Detailed Description	121
6.15.2	Constructor & Destructor Documentation	121
6.15.2.1	__init__()	121
6.15.3	Member Function Documentation	122
6.15.3.1	__str__()	122
6.15.3.2	cacheData()	122
6.15.3.3	checkIfDataExists()	123
6.15.3.4	getConfig()	123
6.15.3.5	getDataLocation()	123
6.15.3.6	getHDFStorage()	124
6.15.3.7	getMetadata()	124
6.15.3.8	multirun_enabled()	125
6.15.3.9	output()	125
6.15.3.10	perturb()	125
6.15.3.11	reset()	125
6.15.3.12	setDataLocation()	125
6.15.3.13	verbose_print()	126
6.15.3.14	writeConfig()	126
6.15.4	Member Data Documentation	126
6.15.4.1	ap_paramList	126
6.15.4.2	llh_url	127
6.15.4.3	memmap	127
6.15.4.4	metadata_url_list	127
6.15.4.5	slc_url_list	127

6.15.4.6	verbose	127
6.16	skdaccess.geo.magnetometer.DataFetcher Class Reference	127
6.16.1	Detailed Description	128
6.16.2	Constructor & Destructor Documentation	128
6.16.2.1	__init__()	129
6.16.3	Member Function Documentation	129
6.16.3.1	__str__()	129
6.16.3.2	getConfig()	129
6.16.3.3	getDataMetadata()	130
6.16.3.4	getMetadata()	130
6.16.3.5	multirun_enabled()	130
6.16.3.6	output()	130
6.16.3.7	perturb()	131
6.16.3.8	reset()	131
6.16.3.9	retrieveOnlineData()	131
6.16.3.10	verbose_print()	131
6.16.3.11	writeConfig()	132
6.16.4	Member Data Documentation	132
6.16.4.1	ap_paramList	132
6.16.4.2	channels	132
6.16.4.3	data_type	132
6.16.4.4	end_time	133
6.16.4.5	interval	133
6.16.4.6	start_time	133
6.16.4.7	verbose	133
6.17	skdaccess.geo.modis.stream.reflectance.DataFetcher Class Reference	133
6.17.1	Detailed Description	134
6.17.2	Constructor & Destructor Documentation	134

6.17.2.1	<code>__init__()</code>	134
6.18	<code>skdaccess.geo.wyoming_sounding.cache.DataFetcher</code> Class Reference	134
6.18.1	Detailed Description	136
6.18.2	Constructor & Destructor Documentation	136
6.18.2.1	<code>__init__()</code>	136
6.18.3	Member Function Documentation	137
6.18.3.1	<code>__str__()</code>	137
6.18.3.2	<code>cacheData()</code>	137
6.18.3.3	<code>checkIfDataExists()</code>	138
6.18.3.4	<code>getConfig()</code>	138
6.18.3.5	<code>getDataLocation()</code>	138
6.18.3.6	<code>getHDFStorage()</code>	139
6.18.3.7	<code>getMetadata()</code>	139
6.18.3.8	<code>multirun_enabled()</code>	139
6.18.3.9	<code>output()</code>	140
6.18.3.10	<code>perturb()</code>	140
6.18.3.11	<code>reset()</code>	140
6.18.3.12	<code>setDataLocation()</code>	140
6.18.3.13	<code>verbose_print()</code>	141
6.18.3.14	<code>writeConfig()</code>	141
6.18.4	Member Data Documentation	141
6.18.4.1	<code>ap_paramList</code>	141
6.18.4.2	<code>day_end</code>	141
6.18.4.3	<code>day_start</code>	142
6.18.4.4	<code>end_hour</code>	142
6.18.4.5	<code>month_list</code>	142
6.18.4.6	<code>start_hour</code>	142
6.18.4.7	<code>station_number</code>	142

6.18.4.8	verbose	142
6.18.4.9	year_list	142
6.19	skdaccess.geo.wyoming_sounding.stream.DataFetcher Class Reference	143
6.19.1	Detailed Description	144
6.19.2	Constructor & Destructor Documentation	144
6.19.2.1	__init__()	144
6.19.3	Member Function Documentation	145
6.19.3.1	__str__()	145
6.19.3.2	getConfig()	145
6.19.3.3	getMetadata()	145
6.19.3.4	multirun_enabled()	145
6.19.3.5	output() [1/2]	146
6.19.3.6	output() [2/2]	146
6.19.3.7	perturb()	146
6.19.3.8	reset()	146
6.19.3.9	retrieveOnlineData()	146
6.19.3.10	verbose_print()	147
6.19.3.11	writeConfig()	147
6.19.4	Member Data Documentation	147
6.19.4.1	ap_paramList	148
6.19.4.2	day_end	148
6.19.4.3	day_start	148
6.19.4.4	end_hour	148
6.19.4.5	month_list	148
6.19.4.6	start_hour	148
6.19.4.7	station_number	148
6.19.4.8	verbose	149
6.19.4.9	year_list	149

6.20	skdaccess.geo.modis.cache.cloud_opacity.DataFetcher Class Reference	149
6.20.1	Detailed Description	149
6.20.2	Constructor & Destructor Documentation	149
6.20.2.1	<code>__init__()</code>	149
6.21	skdaccess.geo.modis.cache.cloud_mask.DataFetcher Class Reference	150
6.21.1	Detailed Description	150
6.21.2	Constructor & Destructor Documentation	150
6.21.2.1	<code>__init__()</code>	151
6.22	skdaccess.geo.modis.cache.reflectance.DataFetcher Class Reference	151
6.22.1	Detailed Description	152
6.22.2	Constructor & Destructor Documentation	152
6.22.2.1	<code>__init__()</code>	152
6.23	skdaccess.geo.modis.cache.DataFetcher Class Reference	152
6.23.1	Detailed Description	154
6.23.2	Constructor & Destructor Documentation	154
6.23.2.1	<code>__init__()</code>	154
6.23.3	Member Function Documentation	155
6.23.3.1	<code>__str__()</code>	155
6.23.3.2	<code>cacheData()</code> [1/2]	155
6.23.3.3	<code>cacheData()</code> [2/2]	156
6.23.3.4	<code>checkIfDataExists()</code>	156
6.23.3.5	<code>find_data()</code>	157
6.23.3.6	<code>getConfig()</code>	157
6.23.3.7	<code>getDataLocation()</code>	157
6.23.3.8	<code>getHDFStorage()</code>	158
6.23.3.9	<code>getMetadata()</code>	158
6.23.3.10	<code>multirun_enabled()</code>	159
6.23.3.11	<code>output()</code>	159

6.23.3.12 perturb()	159
6.23.3.13 reset()	159
6.23.3.14 setDataLocation()	159
6.23.3.15 verbose_print()	160
6.23.3.16 writeConfig()	160
6.23.4 Member Data Documentation	160
6.23.4.1 ap_paramList	160
6.23.4.2 daynightboth	161
6.23.4.3 end_date	161
6.23.4.4 grid	161
6.23.4.5 grid_fill	161
6.23.4.6 modis_id	161
6.23.4.7 modis_identifier	161
6.23.4.8 modis_platform	161
6.23.4.9 start_date	162
6.23.4.10 use_long_name	162
6.23.4.11 variable_list	162
6.23.4.12 verbose	162
6.24 skdaccess.geo.modis.stream.cloud_opacity.DataFetcher Class Reference	162
6.24.1 Detailed Description	163
6.24.2 Constructor & Destructor Documentation	163
6.24.2.1 __init__()	163
6.25 skdaccess.geo.modis.stream.cloud_mask.DataFetcher Class Reference	163
6.25.1 Detailed Description	164
6.25.2 Constructor & Destructor Documentation	164
6.25.2.1 __init__()	164
6.26 skdaccess.planetary.ode.cache.DataFetcher Class Reference	164
6.26.1 Detailed Description	166

6.26.2	Constructor & Destructor Documentation	166
6.26.2.1	__init__()	166
6.26.3	Member Function Documentation	167
6.26.3.1	__str__()	167
6.26.3.2	cacheData()	167
6.26.3.3	checkIfDataExists()	168
6.26.3.4	getConfig()	168
6.26.3.5	getDataLocation()	168
6.26.3.6	getHDFSStorage()	169
6.26.3.7	getMetadata()	169
6.26.3.8	multirun_enabled()	169
6.26.3.9	output()	170
6.26.3.10	perturb()	170
6.26.3.11	reset()	170
6.26.3.12	setDataLocation()	170
6.26.3.13	verbose_print()	171
6.26.3.14	writeConfig()	171
6.26.4	Member Data Documentation	171
6.26.4.1	ap_paramList	171
6.26.4.2	eastern_lon	171
6.26.4.3	file_name	172
6.26.4.4	instrument	172
6.26.4.5	max_lat	172
6.26.4.6	max_ob_time	172
6.26.4.7	min_lat	172
6.26.4.8	min_ob_time	172
6.26.4.9	mission	172
6.26.4.10	number_product_limit	173

6.26.4.11	product_id	173
6.26.4.12	product_type	173
6.26.4.13	remove_ndv	173
6.26.4.14	result_offset_number	173
6.26.4.15	target	173
6.26.4.16	verbose	173
6.26.4.17	western_lon	174
6.27	skdaccess.geo.modis.stream.DataFetcher Class Reference	174
6.27.1	Detailed Description	175
6.27.2	Constructor & Destructor Documentation	175
6.27.2.1	__init__()	175
6.27.3	Member Function Documentation	176
6.27.3.1	__str__()	176
6.27.3.2	getConfig()	176
6.27.3.3	getMetadata()	176
6.27.3.4	multirun_enabled()	177
6.27.3.5	output()	177
6.27.3.6	perturb()	177
6.27.3.7	reset()	177
6.27.3.8	retrieveOnlineData()	177
6.27.3.9	verbose_print()	178
6.27.3.10	writeConfig()	178
6.27.4	Member Data Documentation	178
6.27.4.1	ap_paramList	179
6.27.4.2	daynightboth	179
6.27.4.3	end_date	179
6.27.4.4	grid	179
6.27.4.5	grid_fill	179

6.27.4.6	<code>modis_id</code>	179
6.27.4.7	<code>modis_identifier</code>	179
6.27.4.8	<code>modis_platform</code>	180
6.27.4.9	<code>start_date</code>	180
6.27.4.10	<code>use_long_name</code>	180
6.27.4.11	<code>variable_list</code>	180
6.27.4.12	<code>verbose</code>	180
6.28	<code>skdaccess.geo.grace.mascon.cache.DataFetcher</code> Class Reference	180
6.28.1	Detailed Description	182
6.28.2	Constructor & Destructor Documentation	182
6.28.2.1	<code>__init__()</code>	182
6.28.3	Member Function Documentation	182
6.28.3.1	<code>__str__()</code>	182
6.28.3.2	<code>cacheData()</code>	183
6.28.3.3	<code>checkIfDataExists()</code>	183
6.28.3.4	<code>getConfig()</code>	184
6.28.3.5	<code>getDataLocation()</code>	184
6.28.3.6	<code>getHDFStorage()</code>	184
6.28.3.7	<code>getMasconPlacement()</code>	185
6.28.3.8	<code>getMetadata()</code>	185
6.28.3.9	<code>multirun_enabled()</code>	185
6.28.3.10	<code>output()</code>	186
6.28.3.11	<code>perturb()</code>	186
6.28.3.12	<code>reset()</code>	186
6.28.3.13	<code>setDataLocation()</code>	186
6.28.3.14	<code>verbose_print()</code>	187
6.28.3.15	<code>writeConfig()</code>	187
6.28.4	Member Data Documentation	187

6.28.4.1	ap_paramList	187
6.28.4.2	end_date	187
6.28.4.3	mascon_placement_url	188
6.28.4.4	mascon_url	188
6.28.4.5	scale_factor_url	188
6.28.4.6	start_date	188
6.28.4.7	verbose	188
6.29	skdaccess.geo.imsdnhs.DataFetcher Class Reference	188
6.29.1	Detailed Description	189
6.29.2	Constructor & Destructor Documentation	189
6.29.2.1	__init__()	190
6.29.3	Member Function Documentation	190
6.29.3.1	__str__()	190
6.29.3.2	downloadFullDataset()	190
6.29.3.3	getConfig()	191
6.29.3.4	getDataLocation()	191
6.29.3.5	getMetadata()	191
6.29.3.6	multirun_enabled()	192
6.29.3.7	output()	192
6.29.3.8	perturb()	192
6.29.3.9	reset()	192
6.29.3.10	setDataLocation()	192
6.29.3.11	verbose_print()	193
6.29.3.12	writeConfig()	193
6.29.4	Member Data Documentation	193
6.29.4.1	ap_paramList	193
6.29.4.2	coordinate_dict	194
6.29.4.3	end_date	194

6.29.4.4	start_date	194
6.29.4.5	verbose	194
6.30	skdaccess.geo.era_interim.cache.DataFetcher Class Reference	194
6.30.1	Detailed Description	195
6.30.2	Constructor & Destructor Documentation	196
6.30.2.1	__init__()	196
6.30.3	Member Function Documentation	196
6.30.3.1	__str__()	196
6.30.3.2	cacheData()	196
6.30.3.3	checkIfDataExists()	197
6.30.3.4	getConfig()	197
6.30.3.5	getDataLocation()	198
6.30.3.6	getHDFStorage()	198
6.30.3.7	getMetadata()	198
6.30.3.8	multirun_enabled()	199
6.30.3.9	output()	199
6.30.3.10	perturb()	199
6.30.3.11	reset()	199
6.30.3.12	setDataLocation()	199
6.30.3.13	verbose_print()	200
6.30.3.14	writeConfig()	200
6.30.4	Member Data Documentation	200
6.30.4.1	ap_paramList	200
6.30.4.2	data_names	201
6.30.4.3	date_list	201
6.30.4.4	password	201
6.30.4.5	username	201
6.30.4.6	verbose	201

6.31	skdaccess.geo.ngl_gps.DataFetcher Class Reference	201
6.31.1	Detailed Description	202
6.31.2	Constructor & Destructor Documentation	203
6.31.2.1	__init__()	203
6.31.3	Member Function Documentation	203
6.31.3.1	__str__()	203
6.31.3.2	downloadFullDataset()	203
6.31.3.3	getAntennaLogs()	204
6.31.3.4	getConfig()	204
6.31.3.5	getDataLocation()	204
6.31.3.6	getMetadata()	205
6.31.3.7	getStationMetadata()	205
6.31.3.8	multirun_enabled()	205
6.31.3.9	output()	206
6.31.3.10	perturb()	206
6.31.3.11	reset()	206
6.31.3.12	setDataLocation()	206
6.31.3.13	verbose_print()	207
6.31.3.14	writeConfig()	207
6.31.4	Member Data Documentation	207
6.31.4.1	ap_paramList	207
6.31.4.2	data_type	207
6.31.4.3	end_date	208
6.31.4.4	lat_range	208
6.31.4.5	lon_range	208
6.31.4.6	mdyratio	208
6.31.4.7	start_date	208
6.31.4.8	verbose	208

6.32	skdaccess.geo.mahali.tec.DataFetcher Class Reference	209
6.32.1	Detailed Description	210
6.32.2	Constructor & Destructor Documentation	210
6.32.2.1	__init__()	210
6.32.3	Member Function Documentation	210
6.32.3.1	__str__()	211
6.32.3.2	cacheData()	211
6.32.3.3	checkIfDataExists()	211
6.32.3.4	getConfig()	212
6.32.3.5	getDataLocation()	212
6.32.3.6	getHDFSStorage()	212
6.32.3.7	getMetadata()	213
6.32.3.8	multirun_enabled()	213
6.32.3.9	output()	213
6.32.3.10	perturb()	214
6.32.3.11	reset()	214
6.32.3.12	setDataLocation()	214
6.32.3.13	verbose_print()	214
6.32.3.14	writeConfig()	215
6.32.4	Member Data Documentation	215
6.32.4.1	ap_paramList	215
6.32.4.2	date_range	215
6.32.4.3	end_date	215
6.32.4.4	start_date	216
6.32.4.5	verbose	216
6.33	skdaccess.astro.kepler.DataFetcher Class Reference	216
6.33.1	Detailed Description	217
6.33.2	Constructor & Destructor Documentation	217

6.33.2.1	<code>__init__()</code>	217
6.33.3	Member Function Documentation	218
6.33.3.1	<code>__str__()</code>	218
6.33.3.2	<code>cacheData()</code> [1/2]	218
6.33.3.3	<code>cacheData()</code> [2/2]	218
6.33.3.4	<code>checkIfDataExists()</code>	220
6.33.3.5	<code>downloadKeplerData()</code>	220
6.33.3.6	<code>getConfig()</code>	221
6.33.3.7	<code>getDataLocation()</code>	221
6.33.3.8	<code>getHDFSStorage()</code>	221
6.33.3.9	<code>getMetadata()</code>	222
6.33.3.10	<code>multirun_enabled()</code>	222
6.33.3.11	<code>output()</code>	222
6.33.3.12	<code>perturb()</code>	223
6.33.3.13	<code>reset()</code>	223
6.33.3.14	<code>setDataLocation()</code>	223
6.33.3.15	<code>verbose_print()</code>	223
6.33.3.16	<code>writeConfig()</code>	224
6.33.4	Member Data Documentation	224
6.33.4.1	<code>ap_paramList</code>	224
6.33.4.2	<code>quarter_list</code>	224
6.33.4.3	<code>verbose</code>	224
6.34	<code>skdaccess.geo.pbo.DataFetcher</code> Class Reference	225
6.34.1	Detailed Description	226
6.34.2	Constructor & Destructor Documentation	226
6.34.2.1	<code>__init__()</code>	226
6.34.3	Member Function Documentation	227
6.34.3.1	<code>__str__()</code>	227

6.34.3.2	downloadFullDataset()	227
6.34.3.3	getAntennaLogs()	228
6.34.3.4	getConfig()	228
6.34.3.5	getDataLocation()	228
6.34.3.6	getInfo()	229
6.34.3.7	getMetadata()	229
6.34.3.8	getStationMetadata()	229
6.34.3.9	multirun_enabled()	229
6.34.3.10	output()	230
6.34.3.11	perturb()	230
6.34.3.12	reset()	230
6.34.3.13	setDataLocation()	230
6.34.3.14	setStationList()	231
6.34.3.15	verbose_print()	231
6.34.3.16	writeConfig()	231
6.34.4	Member Data Documentation	232
6.34.4.1	antenna_info	232
6.34.4.2	ap_paramList	232
6.34.4.3	default_columns	232
6.34.4.4	default_error_columns	232
6.34.4.5	index_date_only	232
6.34.4.6	meta_data	232
6.34.4.7	station_list	233
6.34.4.8	use_progress_bar	233
6.34.4.9	verbose	233
6.35	skdaccess.geo.grace.DataFetcher Class Reference	233
6.35.1	Detailed Description	234
6.35.2	Constructor & Destructor Documentation	234

6.35.2.1	<code>__init__()</code>	235
6.35.3	Member Function Documentation	235
6.35.3.1	<code>__str__()</code>	235
6.35.3.2	<code>downloadFullDataset()</code>	235
6.35.3.3	<code>getConfig()</code>	236
6.35.3.4	<code>getDataLocation()</code>	236
6.35.3.5	<code>getMetadata()</code>	236
6.35.3.6	<code>multirun_enabled()</code>	237
6.35.3.7	<code>output()</code>	237
6.35.3.8	<code>perturb()</code>	237
6.35.3.9	<code>reset()</code>	237
6.35.3.10	<code>setDataLocation()</code>	237
6.35.3.11	<code>verbose_print()</code>	238
6.35.3.12	<code>writeConfig()</code>	238
6.35.4	Member Data Documentation	238
6.35.4.1	<code>ap_paramList</code>	238
6.35.4.2	<code>end_date</code>	239
6.35.4.3	<code>start_date</code>	239
6.35.4.4	<code>verbose</code>	239
6.36	<code>skdaccess.geo.mahali.rinex.DataFetcher</code> Class Reference	239
6.36.1	Detailed Description	241
6.36.2	Constructor & Destructor Documentation	241
6.36.2.1	<code>__init__()</code>	241
6.36.3	Member Function Documentation	241
6.36.3.1	<code>__str__()</code>	241
6.36.3.2	<code>cacheData()</code> [1/2]	241
6.36.3.3	<code>cacheData()</code> [2/2]	242
6.36.3.4	<code>checkIfDataExists()</code>	242

6.36.3.5	getConfig()	243
6.36.3.6	getDataLocation()	243
6.36.3.7	getHDFSStorage()	243
6.36.3.8	getMetadata()	244
6.36.3.9	multirun_enabled()	244
6.36.3.10	output()	244
6.36.3.11	perturb()	245
6.36.3.12	reset()	245
6.36.3.13	setDataLocation()	245
6.36.3.14	verbose_print()	245
6.36.3.15	writeConfig()	246
6.36.4	Member Data Documentation	246
6.36.4.1	ap_paramList	246
6.36.4.2	date_range	246
6.36.4.3	end_date	246
6.36.4.4	generate_links	247
6.36.4.5	start_date	247
6.36.4.6	verbose	247
6.37	skdaccess.geo.mahali.temperature.DataFetcher Class Reference	247
6.37.1	Detailed Description	248
6.37.2	Constructor & Destructor Documentation	248
6.37.2.1	__init__()	248
6.37.3	Member Function Documentation	249
6.37.3.1	__str__()	249
6.37.3.2	getConfig()	249
6.37.3.3	getMetadata()	249
6.37.3.4	multirun_enabled()	250
6.37.3.5	output()	250

6.37.3.6	perturb()	250
6.37.3.7	reset()	250
6.37.3.8	retrieveOnlineData()	250
6.37.3.9	verbose_print()	251
6.37.3.10	writeConfig()	251
6.37.4	Member Data Documentation	251
6.37.4.1	ap_paramList	252
6.37.4.2	end_date	252
6.37.4.3	start_date	252
6.37.4.4	verbose	252
6.38	skdaccess.astro.voyager.DataFetcher Class Reference	252
6.38.1	Detailed Description	254
6.38.2	Constructor & Destructor Documentation	254
6.38.2.1	__init__()	254
6.38.3	Member Function Documentation	254
6.38.3.1	__str__()	254
6.38.3.2	cacheData()	255
6.38.3.3	checkIfDataExists()	255
6.38.3.4	generateURL()	256
6.38.3.5	getConfig()	256
6.38.3.6	getDataLocation()	256
6.38.3.7	getHDFStorage()	257
6.38.3.8	getMetadata()	257
6.38.3.9	getMetadataFiles()	258
6.38.3.10	multirun_enabled()	258
6.38.3.11	output()	258
6.38.3.12	parseVoyagerData()	258
6.38.3.13	parseVoyagerMetadata()	259

6.38.3.14 perturb()	259
6.38.3.15 reset()	259
6.38.3.16 setDataLocation()	260
6.38.3.17 verbose_print()	260
6.38.3.18 writeConfig()	260
6.38.4 Member Data Documentation	261
6.38.4.1 ap_paramList	261
6.38.4.2 base_url	261
6.38.4.3 field_names	261
6.38.4.4 field_widths	261
6.38.4.5 spacecraft_list	261
6.38.4.6 verbose	261
6.38.4.7 year_list	262
6.39 skdaccess.solar.sdo.DataFetcher Class Reference	262
6.39.1 Detailed Description	263
6.39.2 Constructor & Destructor Documentation	263
6.39.2.1 __init__()	263
6.39.3 Member Function Documentation	263
6.39.3.1 __str__()	263
6.39.3.2 getConfig()	264
6.39.3.3 getMetadata()	264
6.39.3.4 multirun_enabled()	264
6.39.3.5 output()	264
6.39.3.6 perturb()	265
6.39.3.7 reset()	265
6.39.3.8 retrieveOnlineData()	265
6.39.3.9 verbose_print()	265
6.39.3.10 writeConfig()	266

6.39.4	Member Data Documentation	266
6.39.4.1	ap_paramList	266
6.39.4.2	verbose	266
6.40	skdaccess.framework.data_class.DataFetcherBase Class Reference	267
6.40.1	Detailed Description	267
6.40.2	Constructor & Destructor Documentation	268
6.40.2.1	__init__()	268
6.40.3	Member Function Documentation	268
6.40.3.1	__str__()	268
6.40.3.2	getConfig()	268
6.40.3.3	getMetadata()	269
6.40.3.4	multirun_enabled()	269
6.40.3.5	output()	269
6.40.3.6	perturb()	269
6.40.3.7	reset()	270
6.40.3.8	verbose_print()	270
6.40.3.9	writeConfig()	270
6.40.4	Member Data Documentation	270
6.40.4.1	ap_paramList	270
6.40.4.2	verbose	271
6.41	skdaccess.framework.data_class.DataFetcherCache Class Reference	271
6.41.1	Detailed Description	272
6.41.2	Member Function Documentation	272
6.41.2.1	__str__()	272
6.41.2.2	cacheData()	273
6.41.2.3	checkIfDataExists()	273
6.41.2.4	getConfig()	274
6.41.2.5	getDataLocation()	274

6.41.2.6	getHDFSStorage()	274
6.41.2.7	getMetadata()	275
6.41.2.8	multirun_enabled()	275
6.41.2.9	output()	275
6.41.2.10	perturb()	276
6.41.2.11	reset()	276
6.41.2.12	setDataLocation()	276
6.41.2.13	verbose_print()	276
6.41.2.14	writeConfig()	277
6.41.3	Member Data Documentation	277
6.41.3.1	ap_paramList	277
6.41.3.2	verbose	277
6.42	skdaccess.framework.data_class.DataFetcherLocal Class Reference	278
6.42.1	Detailed Description	279
6.42.2	Member Function Documentation	279
6.42.2.1	__str__()	279
6.42.2.2	getConfig()	279
6.42.2.3	getDataLocation()	279
6.42.2.4	getMetadata()	280
6.42.2.5	multirun_enabled()	280
6.42.2.6	output()	280
6.42.2.7	perturb()	281
6.42.2.8	reset()	281
6.42.2.9	setDataLocation()	281
6.42.2.10	verbose_print()	281
6.42.2.11	writeConfig()	282
6.42.3	Member Data Documentation	282
6.42.3.1	ap_paramList	282

6.42.3.2	verbose	282
6.43	skdaccess.framework.data_class.DataFetcherStorage Class Reference	282
6.43.1	Detailed Description	283
6.43.2	Member Function Documentation	283
6.43.2.1	__str__()	284
6.43.2.2	downloadFullDataset()	284
6.43.2.3	getConfig()	284
6.43.2.4	getDataLocation()	284
6.43.2.5	getMetadata()	285
6.43.2.6	multirun_enabled()	285
6.43.2.7	output()	285
6.43.2.8	perturb()	286
6.43.2.9	reset()	286
6.43.2.10	setDataLocation()	286
6.43.2.11	verbose_print()	286
6.43.2.12	writeConfig()	287
6.43.3	Member Data Documentation	287
6.43.3.1	ap_paramList	287
6.43.3.2	verbose	287
6.44	skdaccess.framework.data_class.DataFetcherStream Class Reference	287
6.44.1	Detailed Description	288
6.44.2	Member Function Documentation	288
6.44.2.1	__str__()	288
6.44.2.2	getConfig()	289
6.44.2.3	getMetadata()	289
6.44.2.4	multirun_enabled()	289
6.44.2.5	output()	289
6.44.2.6	perturb()	290

6.44.2.7	reset()	290
6.44.2.8	retrieveOnlineData()	290
6.44.2.9	verbose_print()	290
6.44.2.10	writeConfig()	291
6.44.3	Member Data Documentation	291
6.44.3.1	ap_paramList	291
6.44.3.2	verbose	291
6.45	skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper Class Reference	292
6.45.1	Detailed Description	293
6.45.2	Member Function Documentation	293
6.45.2.1	__len__()	293
6.45.2.2	addResult()	293
6.45.2.3	get()	293
6.45.2.4	getIterator()	294
6.45.2.5	getResults()	294
6.45.2.6	getRunID()	294
6.45.2.7	info()	295
6.45.2.8	reset()	295
6.45.2.9	update()	295
6.45.2.10	updateMetadata()	295
6.45.3	Member Data Documentation	296
6.45.3.1	constants	296
6.45.3.2	data	296
6.45.3.3	meta_data	296
6.45.3.4	results	296
6.45.3.5	run_id	296
6.46	skdaccess.framework.data_class.DataWrapperBase Class Reference	297
6.46.1	Detailed Description	297

6.46.2	Constructor & Destructor Documentation	298
6.46.2.1	__init__()	298
6.46.3	Member Function Documentation	298
6.46.3.1	__len__()	298
6.46.3.2	addResult()	298
6.46.3.3	get()	299
6.46.3.4	getIterator()	299
6.46.3.5	getResults()	299
6.46.3.6	getRunID()	300
6.46.3.7	info()	300
6.46.3.8	reset()	300
6.46.3.9	update()	300
6.46.3.10	updateMetadata()	301
6.46.4	Member Data Documentation	301
6.46.4.1	constants	301
6.46.4.2	data	301
6.46.4.3	meta_data	301
6.46.4.4	results	302
6.46.4.5	run_id	302
6.47	skdaccess.utilities.file_browser.FileBrowser Class Reference	302
6.47.1	Constructor & Destructor Documentation	302
6.47.1.1	__init__()	302
6.47.2	Member Function Documentation	303
6.47.2.1	widget()	303
6.47.3	Member Data Documentation	303
6.47.3.1	dirs	303
6.47.3.2	files	303
6.47.3.3	path	303

6.48	skdaccess.framework.data_class.ImageWrapper Class Reference	303
6.48.1	Detailed Description	304
6.48.2	Member Function Documentation	304
6.48.2.1	__len__()	305
6.48.2.2	addResult()	305
6.48.2.3	deleteData()	305
6.48.2.4	get()	305
6.48.2.5	getIterator()	306
6.48.2.6	getResults()	306
6.48.2.7	getRunID()	306
6.48.2.8	info()	307
6.48.2.9	reset()	307
6.48.2.10	update()	307
6.48.2.11	updateData()	307
6.48.2.12	updateMetadata()	308
6.48.3	Member Data Documentation	308
6.48.3.1	constants	308
6.48.3.2	data	308
6.48.3.3	meta_data	308
6.48.3.4	results	309
6.48.3.5	run_id	309
6.49	skdaccess.utilities.modis_util.LatLon Class Reference	309
6.49.1	Detailed Description	310
6.49.2	Constructor & Destructor Documentation	310
6.49.2.1	__init__()	310
6.49.3	Member Function Documentation	310
6.49.3.1	__call__()	310
6.49.4	Member Data Documentation	311

6.49.4.1	alat	311
6.49.4.2	alon	311
6.49.4.3	lat_data	311
6.49.4.4	lon_data	311
6.49.4.5	x_offset	311
6.49.4.6	y_offset	311
6.50	skdaccess.utilities.image_util.LinearGeolocation Class Reference	312
6.50.1	Detailed Description	312
6.50.2	Constructor & Destructor Documentation	313
6.50.2.1	__init__()	313
6.50.3	Member Function Documentation	313
6.50.3.1	getExtents()	313
6.50.3.2	getLatLon()	313
6.50.3.3	getYX()	314
6.50.4	Member Data Documentation	314
6.50.4.1	flip_y	314
6.50.4.2	lat_extents	314
6.50.4.3	lat_pixel_size	315
6.50.4.4	len_x	315
6.50.4.5	len_y	315
6.50.4.6	lon_extents	315
6.50.4.7	lon_pixel_size	315
6.50.4.8	start_lat	315
6.50.4.9	start_lon	315
6.50.4.10	x_offset	316
6.50.4.11	y_offset	316
6.51	skdaccess.framework.data_class.SeriesDictionaryWrapper Class Reference	316
6.51.1	Detailed Description	317

6.51.2 Member Function Documentation	317
6.51.2.1 __len__()	317
6.51.2.2 addResult()	317
6.51.2.3 get()	318
6.51.2.4 getIndices()	318
6.51.2.5 getIterator()	318
6.51.2.6 getLength()	319
6.51.2.7 getResults()	319
6.51.2.8 getRunID()	319
6.51.2.9 info()	319
6.51.2.10 reset()	320
6.51.2.11 update()	320
6.51.2.12 updateMetadata()	320
6.51.3 Member Data Documentation	320
6.51.3.1 constants	320
6.51.3.2 data	321
6.51.3.3 data_names	321
6.51.3.4 error_names	321
6.51.3.5 meta_data	321
6.51.3.6 results	321
6.51.3.7 run_id	321
6.52 skdaccess.framework.data_class.SeriesWrapper Class Reference	322
6.52.1 Detailed Description	323
6.52.2 Constructor & Destructor Documentation	323
6.52.2.1 __init__()	323
6.52.3 Member Function Documentation	323
6.52.3.1 __len__()	324
6.52.3.2 addResult()	324

6.52.3.3	get()	324
6.52.3.4	getIndices()	325
6.52.3.5	getIterator()	325
6.52.3.6	getLength()	325
6.52.3.7	getResults()	325
6.52.3.8	getRunID()	326
6.52.3.9	info()	326
6.52.3.10	reset()	326
6.52.3.11	update()	326
6.52.3.12	updateMetadata()	327
6.52.4	Member Data Documentation	327
6.52.4.1	constants	327
6.52.4.2	data	327
6.52.4.3	data_names	327
6.52.4.4	error_names	328
6.52.4.5	meta_data	328
6.52.4.6	results	328
6.52.4.7	run_id	328
6.53	skdaccess.utilities.sounding_util.SoundingParser Class Reference	328
6.53.1	Detailed Description	329
6.53.2	Constructor & Destructor Documentation	329
6.53.2.1	__init__()	329
6.53.3	Member Function Documentation	329
6.53.3.1	handle_data()	329
6.53.3.2	handle_endtag()	330
6.53.3.3	handle_starttag()	330
6.53.4	Member Data Documentation	330
6.53.4.1	data_dict	330

6.53.4.2	in_header	331
6.53.4.3	in_pre_tag	331
6.53.4.4	label	331
6.53.4.5	metadata_dict	331
6.53.4.6	read_data	331
6.53.4.7	tmp	331
6.54	skdaccess.utilities.image_util.SplineLatLon Class Reference	332
6.54.1	Detailed Description	332
6.54.2	Constructor & Destructor Documentation	332
6.54.2.1	__init__()	333
6.54.3	Member Function Documentation	333
6.54.3.1	__call__()	333
6.54.4	Member Data Documentation	334
6.54.4.1	lat_func	334
6.54.4.2	lon_func	334
6.54.4.3	x_offset	334
6.54.4.4	y_offset	334
6.55	skdaccess.framework.data_class.TableWrapper Class Reference	335
6.55.1	Detailed Description	336
6.55.2	Constructor & Destructor Documentation	336
6.55.2.1	__init__()	336
6.55.3	Member Function Documentation	337
6.55.3.1	__len__()	337
6.55.3.2	addColumn()	337
6.55.3.3	addResult()	337
6.55.3.4	get()	338
6.55.3.5	getDefaultColumns()	338
6.55.3.6	getDefaultErrorColumns()	338

6.55.3.7	getIterator()	339
6.55.3.8	getLength()	339
6.55.3.9	getResults()	339
6.55.3.10	getRunID()	339
6.55.3.11	info()	340
6.55.3.12	removeFrames()	340
6.55.3.13	reset()	340
6.55.3.14	update()	340
6.55.3.15	updateData()	341
6.55.3.16	updateFrames()	341
6.55.3.17	updateMetadata()	341
6.55.4	Member Data Documentation	342
6.55.4.1	constants	342
6.55.4.2	data	342
6.55.4.3	default_columns	342
6.55.4.4	default_error_columns	342
6.55.4.5	meta_data	342
6.55.4.6	results	343
6.55.4.7	run_id	343
6.56	skdaccess.framework.data_class.XArrayWrapper Class Reference	343
6.56.1	Detailed Description	344
6.56.2	Constructor & Destructor Documentation	344
6.56.2.1	__init__()	344
6.56.3	Member Function Documentation	344
6.56.3.1	__len__()	344
6.56.3.2	addResult()	344
6.56.3.3	get()	345
6.56.3.4	getIterator()	345

6.56.3.5	getResults()	345
6.56.3.6	getRunID()	346
6.56.3.7	info()	346
6.56.3.8	reset()	346
6.56.3.9	update()	346
6.56.3.10	updateMetadata()	347
6.56.4	Member Data Documentation	347
6.56.4.1	constants	347
6.56.4.2	data	347
6.56.4.3	index_list	347
6.56.4.4	meta_data	348
6.56.4.5	results	348
6.56.4.6	run_id	348
7	File Documentation	349
7.1	framework/data_class.py File Reference	349
7.2	framework/param_class.py File Reference	350
7.3	geo/mahali/rinex/data_wrapper.py File Reference	350
7.4	solar/sdo/data_fetcher.py File Reference	350
7.5	planetary/ode/cache/data_fetcher.py File Reference	351
7.6	geo/grace/mascon/cache/data_fetcher.py File Reference	351
7.7	geo/grace/data_fetcher.py File Reference	351
7.8	geo/mahali/tec/data_fetcher.py File Reference	352
7.9	geo/mahali/rinex/data_fetcher.py File Reference	352
7.10	geo/mahali/temperature/data_fetcher.py File Reference	352
7.11	geo/ngl_gps/data_fetcher.py File Reference	352
7.12	geo/era_interim/cache/data_fetcher.py File Reference	353
7.13	geo/imsdnhs/data_fetcher.py File Reference	353

7.14	geo/gldas/data_fetcher.py File Reference	353
7.15	geo/sentinel_1/cache/data_fetcher.py File Reference	354
7.16	geo/magnetometer/data_fetcher.py File Reference	354
7.17	geo/wyoming_sounding/cache/data_fetcher.py File Reference	354
7.18	geo/wyoming_sounding/stream/data_fetcher.py File Reference	354
7.19	geo/modis/cache/cloud_opacity/data_fetcher.py File Reference	355
7.20	geo/modis/cache/cloud_mask/data_fetcher.py File Reference	355
7.21	geo/modis/cache/reflectance/data_fetcher.py File Reference	355
7.22	geo/modis/cache/data_fetcher.py File Reference	356
7.23	geo/modis/stream/cloud_opacity/data_fetcher.py File Reference	356
7.24	geo/modis/stream/cloud_mask/data_fetcher.py File Reference	356
7.25	geo/modis/stream/reflectance/data_fetcher.py File Reference	356
7.26	geo/modis/stream/data_fetcher.py File Reference	357
7.27	geo/uavsar/cache/data_fetcher.py File Reference	357
7.28	geo/srtm/cache/data_fetcher.py File Reference	357
7.29	geo/groundwater/data_fetcher.py File Reference	358
7.30	geo/pbo/data_fetcher.py File Reference	358
7.31	astro/kepler/data_fetcher.py File Reference	358
7.32	astro/voyager/data_fetcher.py File Reference	358
7.33	utilities/file_browser.py File Reference	359
7.34	utilities/grace_util.py File Reference	359
7.35	utilities/gw_util.py File Reference	359
7.36	utilities/image_util.py File Reference	360
7.37	utilities/kepler_util.py File Reference	360
7.38	utilities/mahali_util.py File Reference	361
7.39	utilities/modis_util.py File Reference	361
7.40	utilities/ode_util.py File Reference	362
7.41	utilities/pbo_util.py File Reference	362
7.42	utilities/sentinel_1_util.py File Reference	363
7.43	utilities/sounding_util.py File Reference	363
7.44	utilities/srtm_util.py File Reference	363
7.45	utilities/support.py File Reference	364
7.46	utilities/uavsar_util.py File Reference	364

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	13
skdaccess.astro.voyager	14
skdaccess.astro.voyager.data_fetcher	14
skdaccess.framework	14
skdaccess.framework.data_class	14
skdaccess.framework.param_class	15
skdaccess.geo	15
skdaccess.geo.era_interim	15
skdaccess.geo.era_interim.cache	16
skdaccess.geo.era_interim.cache.data_fetcher	16
skdaccess.geo.gldas	16
skdaccess.geo.gldas.data_fetcher	16
skdaccess.geo.grace	16
skdaccess.geo.grace.data_fetcher	16
skdaccess.geo.grace.mascon	17
skdaccess.geo.grace.mascon.cache	17
skdaccess.geo.grace.mascon.cache.data_fetcher	17
skdaccess.geo.groundwater	17
skdaccess.geo.groundwater.data_fetcher	17
skdaccess.geo.imsdnhs	17
skdaccess.geo.imsdnhs.data_fetcher	18
skdaccess.geo.magnetometer	18
skdaccess.geo.magnetometer.data_fetcher	18
skdaccess.geo.mahali	18
skdaccess.geo.mahali.rinex	18
skdaccess.geo.mahali.rinex.data_fetcher	19
skdaccess.geo.mahali.rinex.data_wrapper	19
skdaccess.geo.mahali.tec	19

skdaccess.geo.mahali.tec.data_fetcher	19
skdaccess.geo.mahali.temperature	19
skdaccess.geo.mahali.temperature.data_fetcher	19
skdaccess.geo.modis	20
skdaccess.geo.modis.cache	20
skdaccess.geo.modis.cache.cloud_mask	20
skdaccess.geo.modis.cache.cloud_mask.data_fetcher	20
skdaccess.geo.modis.cache.cloud_opacity	20
skdaccess.geo.modis.cache.cloud_opacity.data_fetcher	20
skdaccess.geo.modis.cache.data_fetcher	21
skdaccess.geo.modis.cache.reflectance	21
skdaccess.geo.modis.cache.reflectance.data_fetcher	21
skdaccess.geo.modis.stream	21
skdaccess.geo.modis.stream.cloud_mask	21
skdaccess.geo.modis.stream.cloud_mask.data_fetcher	21
skdaccess.geo.modis.stream.cloud_opacity	22
skdaccess.geo.modis.stream.cloud_opacity.data_fetcher	22
skdaccess.geo.modis.stream.data_fetcher	22
skdaccess.geo.modis.stream.reflectance	22
skdaccess.geo.modis.stream.reflectance.data_fetcher	22
skdaccess.geo.ngl_gps	22
skdaccess.geo.ngl_gps.data_fetcher	23
skdaccess.geo.pbo	23
skdaccess.geo.pbo.data_fetcher	23
skdaccess.geo.sentinel_1	23
skdaccess.geo.sentinel_1.cache	23
skdaccess.geo.sentinel_1.cache.data_fetcher	23
skdaccess.geo.srtm	24
skdaccess.geo.srtm.cache	24
skdaccess.geo.srtm.cache.data_fetcher	24
skdaccess.geo.uavsar	24
skdaccess.geo.uavsar.cache	24
skdaccess.geo.uavsar.cache.data_fetcher	24
skdaccess.geo.wyoming_sounding	25
skdaccess.geo.wyoming_sounding.cache	25
skdaccess.geo.wyoming_sounding.cache.data_fetcher	25
skdaccess.geo.wyoming_sounding.stream	25
skdaccess.geo.wyoming_sounding.stream.data_fetcher	25
skdaccess.planetary	25
skdaccess.planetary.ode	26
skdaccess.planetary.ode.cache	26
skdaccess.planetary.ode.cache.data_fetcher	26
skdaccess.solar	26
skdaccess.solar.sdo	26
skdaccess.solar.sdo.data_fetcher	26
skdaccess.utilities	27
skdaccess.utilities.file_browser	27
skdaccess.utilities.grace_util	27
skdaccess.utilities.gw_util	31
skdaccess.utilities.image_util	31
skdaccess.utilities.kepler_util	35
skdaccess.utilities.mahali_util	35
skdaccess.utilities.modis_util	36
skdaccess.utilities.ode_util	41

skdaccess.utilities.pbo_util	45
skdaccess.utilities.sentinel_1_util	49
skdaccess.utilities.sounding_util	50
skdaccess.utilities.srtm_util	51
skdaccess.utilities.support	52
skdaccess.utilities.uavsar_util	53

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

skdaccess.framework.param_class.AutoParam	78
skdaccess.framework.param_class.AutoParamList	81
skdaccess.framework.param_class.AutoParamListCycle	84
skdaccess.framework.param_class.AutoParamMinMax	87
MDF	
skdaccess.geo.modis.cache.cloud_mask.DataFetcher	150
skdaccess.geo.modis.cache.cloud_opacity.DataFetcher	149
skdaccess.geo.modis.cache.reflectance.DataFetcher	151
skdaccess.geo.modis.stream.cloud_mask.DataFetcher	163
skdaccess.geo.modis.stream.cloud_opacity.DataFetcher	162
skdaccess.geo.modis.stream.reflectance.DataFetcher	133
object	
skdaccess.framework.data_class.DataFetcherBase	267
skdaccess.framework.data_class.DataFetcherLocal	278
skdaccess.framework.data_class.DataFetcherCache	271
skdaccess.astro.kepler.DataFetcher	216
skdaccess.astro.voyager.DataFetcher	252
skdaccess.geo.era_interim.cache.DataFetcher	194
skdaccess.geo.grace.mascon.cache.DataFetcher	180
skdaccess.geo.mahali.rinex.DataFetcher	239
skdaccess.geo.mahali.tec.DataFetcher	209
skdaccess.geo.modis.cache.DataFetcher	152
skdaccess.geo.sentinel_1.cache.DataFetcher	96
skdaccess.geo.srtm.cache.DataFetcher	111
skdaccess.geo.uavsar.cache.DataFetcher	120
skdaccess.geo.wyoming_sounding.cache.DataFetcher	134
skdaccess.planetary.ode.cache.DataFetcher	164
skdaccess.framework.data_class.DataFetcherStorage	282
skdaccess.geo.gldas.DataFetcher	90
skdaccess.geo.grace.DataFetcher	233

skdaccess.geo.groundwater.DataFetcher	105
skdaccess.geo.imsdnhs.DataFetcher	188
skdaccess.geo.ngl_gps.DataFetcher	201
skdaccess.geo.pbo.DataFetcher	225
skdaccess.framework.data_class.DataFetcherStream	287
skdaccess.geo.magnetometer.DataFetcher	127
skdaccess.geo.mahali.temperature.DataFetcher	247
skdaccess.geo.modis.stream.DataFetcher	174
skdaccess.geo.wyoming_sounding.stream.DataFetcher	143
skdaccess.solar.sdo.DataFetcher	262
skdaccess.framework.data_class.DataWrapperBase	297
skdaccess.framework.data_class.ImageWrapper	303
skdaccess.framework.data_class.SeriesWrapper	322
skdaccess.framework.data_class.SeriesDictionaryWrapper	316
skdaccess.framework.data_class.TableWrapper	335
skdaccess.framework.data_class.XArrayWrapper	343
skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper	292
skdaccess.framework.param_class.AutoList	57
skdaccess.framework.param_class.AutoListCycle	61
skdaccess.framework.param_class.AutoListPermute	66
skdaccess.framework.param_class.AutoListRemove	70
skdaccess.framework.param_class.AutoListSubset	74
skdaccess.utilities.file_browser.FileBrowser	302
skdaccess.utilities.image_util.AffineGlobalCoords	55
skdaccess.utilities.image_util.LinearGeolocation	312
skdaccess.utilities.image_util.SplineLatLon	332
skdaccess.utilities.modis_util.LatLon	309
HTMLParser	
skdaccess.utilities.sounding_util.SoundingParser	328

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	55
skdaccess.framework.param_class.AutoList	
Specifies a list for returning selections of lists, as opposed to a single element	57
skdaccess.framework.param_class.AutoListCycle	
An Autolist that cycles through different lists	61
skdaccess.framework.param_class.AutoListPermute	
A perturber that permutes a list	66
skdaccess.framework.param_class.AutoListRemove	
Removes a different single element from the initial list at each perturb call	70
skdaccess.framework.param_class.AutoListSubset	
An AutoList perturber that creates random subsets of a list	74
skdaccess.framework.param_class.AutoParam	
Defines a tunable parameter class inherited by specific subclasses	78
skdaccess.framework.param_class.AutoParamList	
A tunable parameter with a specified list of choices that can be randomly selected via perturb	81
skdaccess.framework.param_class.AutoParamListCycle	
Cycles through a list of paramters	84
skdaccess.framework.param_class.AutoParamMinMax	
A tunable parameter with min and max ranges, perturbs to a random value in range	87
skdaccess.geo.gldas.DataFetcher	
Data Fetcher for GLDAS data	90
skdaccess.geo.sentinel_1.cache.DataFetcher	
DataFetcher for retrieving Sentinel SLC data	96
skdaccess.geo.groundwater.DataFetcher	
Generates Data Wrappers of groundwater measurements taken in the US	105
skdaccess.geo.srtm.cache.DataFetcher	
DataFetcher for retrieving data from the Shuttle Radar Topography Mission	111
skdaccess.geo.uavsar.cache.DataFetcher	
Data Fetcher for UAVSAR data	120
skdaccess.geo.magnetometer.DataFetcher	
Data fetcher for USGS geomagnetic observatories	127

skdaccess.geo.modis.stream.reflectance.DataFetcher	
Data fetcher for the modis surface reflectance product ('09', 1 km resolution)	133
skdaccess.geo.wyoming_sounding.cache.DataFetcher	
DataFetcher for retrieving Wyoming Sounding data	134
skdaccess.geo.wyoming_sounding.stream.DataFetcher	
DataFetcher for retrieving Wyoming Sounding data	143
skdaccess.geo.modis.cache.cloud_opacity.DataFetcher	
Data Fetcher for MODIS Cloud Opacity	149
skdaccess.geo.modis.cache.cloud_mask.DataFetcher	
Data Fetcher for MODIS Cloud Mask	150
skdaccess.geo.modis.cache.reflectance.DataFetcher	
Data fetcher for the modis surface reflectance product ('09', 1 km resolution)	151
skdaccess.geo.modis.cache.DataFetcher	
Data Fetcher for MODIS data	152
skdaccess.geo.modis.stream.cloud_opacity.DataFetcher	
Data Fetcher for MODIS Cloud Opacity	162
skdaccess.geo.modis.stream.cloud_mask.DataFetcher	
Data Fetcher for MODIS Cloud Mask	163
skdaccess.planetary.ode.cache.DataFetcher	
Data Fetcher from the Orbital Data Explorer (ODE)	164
skdaccess.geo.modis.stream.DataFetcher	
Data Fetcher for MODIS data	174
skdaccess.geo.grace.mascon.cache.DataFetcher	
Data Fetcher for GRACE mascon data	180
skdaccess.geo.imsdnhs.DataFetcher	
Fetches data for the Interactive Multisensor Snow and Ice Mapping System Daily Northern Hemisphere Snow and Ice Analysis	188
skdaccess.geo.era_interim.cache.DataFetcher	
DataFetcher for retrieving ERA-I data	194
skdaccess.geo.ngl_gps.DataFetcher	
Data fetcher for GPS data from Nevada Geodetic Laboratory	201
skdaccess.geo.mahali.tec.DataFetcher	
Data Fetcher for Mahali Data	209
skdaccess.astro.kepler.DataFetcher	
Data Fetcher for Kepler light curve data	216
skdaccess.geo.pbo.DataFetcher	
Data fetcher for PBO GPS data	225
skdaccess.geo.grace.DataFetcher	
Data Fetcher for GRACE data	233
skdaccess.geo.mahali.rinex.DataFetcher	
Data Fetcher for Mahali Data	239
skdaccess.geo.mahali.temperature.DataFetcher	
Data Fetcher for Mahali temperature data	247
skdaccess.astro.voyager.DataFetcher	
Data Fetcher for Mahali temperature data	252
skdaccess.solar.sdo.DataFetcher	
Data Fetcher for Mahali temperature data	262
skdaccess.framework.data_class.DataFetcherBase	
Base class for all data fetchers	267
skdaccess.framework.data_class.DataFetcherCache	
Data fetcher base class for downloading data and caching results on hard disk	271
skdaccess.framework.data_class.DataFetcherLocal	
Data fetcher base class for use when storing data locally	278

skdaccess.framework.data_class.DataFetcherStorage	
Data fetcher base class for use when entire data set is downloaded	282
skdaccess.framework.data_class.DataFetcherStream	
Data fetcher base class for downloading data into memory	287
skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper	
Data wrapper for Mahali data	292
skdaccess.framework.data_class.DataWrapperBase	
Base class for wrapping data for use in DiscoveryPipeline	297
skdaccess.utilities.file_browser.FileBrowser	
	302
skdaccess.framework.data_class.ImageWrapper	
Wrapper for image data	303
skdaccess.utilities.modis_util.LatLon	
Calculates Lat/Lon position from y,x pixel coordinate	309
skdaccess.utilities.image_util.LinearGeolocation	
This class provides functions to convert between pixel and geodetic coordinates	312
skdaccess.framework.data_class.SeriesDictionaryWrapper	
Data wrapper for series data using a dictionary of data frames	316
skdaccess.framework.data_class.SeriesWrapper	
Data wrapper for series data using a data panel	322
skdaccess.utilities.sounding_util.SoundingParser	
This class parses Wyoming Sounding data	328
skdaccess.utilities.image_util.SplineLatLon	
Holds a 2d spline for interpolating lat/lon grid	332
skdaccess.framework.data_class.TableWrapper	
Data wrapper for table data using an ordered dictionary	335
skdaccess.framework.data_class.XArrayWrapper	
Wrapper for xarrays	343

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

astro/kepler/ data_fetcher.py	358
astro/voyager/ data_fetcher.py	358
framework/ data_class.py	349
framework/ param_class.py	350
geo/era_interim/cache/ data_fetcher.py	353
geo/gldas/ data_fetcher.py	353
geo/grace/ data_fetcher.py	351
geo/grace/mascon/cache/ data_fetcher.py	351
geo/groundwater/ data_fetcher.py	358
geo/imsdnhs/ data_fetcher.py	353
geo/magnetometer/ data_fetcher.py	354
geo/mahali/rinex/ data_fetcher.py	352
geo/mahali/rinex/ data_wrapper.py	350
geo/mahali/tec/ data_fetcher.py	352
geo/mahali/temperature/ data_fetcher.py	352
geo/modis/cache/ data_fetcher.py	356
geo/modis/cache/cloud_mask/ data_fetcher.py	355
geo/modis/cache/cloud_opacity/ data_fetcher.py	355
geo/modis/cache/reflectance/ data_fetcher.py	355
geo/modis/stream/ data_fetcher.py	357
geo/modis/stream/cloud_mask/ data_fetcher.py	356
geo/modis/stream/cloud_opacity/ data_fetcher.py	356
geo/modis/stream/reflectance/ data_fetcher.py	356
geo/ngl_gps/ data_fetcher.py	352
geo/pbo/ data_fetcher.py	358
geo/sentinel_1/cache/ data_fetcher.py	354
geo/srtm/cache/ data_fetcher.py	357
geo/uavsar/cache/ data_fetcher.py	357
geo/wyoming_sounding/cache/ data_fetcher.py	354
geo/wyoming_sounding/stream/ data_fetcher.py	354
planetary/ode/cache/ data_fetcher.py	351

solar/sdo/data_fetcher.py	350
utilities/file_browser.py	359
utilities/grace_util.py	359
utilities/gw_util.py	359
utilities/image_util.py	360
utilities/kepler_util.py	360
utilities/mahali_util.py	361
utilities/modis_util.py	361
utilities/ode_util.py	362
utilities/pbo_util.py	362
utilities/sentinel_1_util.py	363
utilities/sounding_util.py	363
utilities/srtm_util.py	363
utilities/support.py	364
utilities/uavsar_util.py	364

Chapter 5

Namespace Documentation

5.1 skdaccess Namespace Reference

Namespaces

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

5.2 skdaccess.astro Namespace Reference

Namespaces

- [kepler](#)
- [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.voyager Namespace Reference

Namespaces

- [data_fetcher](#)

5.6 skdaccess.astro.voyager.data_fetcher Namespace Reference

Classes

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

5.7 skdaccess.framework Namespace Reference

Namespaces

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

5.8 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.

5.9 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.

5.10 skdaccess.geo Namespace Reference

Namespaces

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

5.11 skdaccess.geo.era_interim Namespace Reference

Namespaces

- [cache](#)

5.12 skdaccess.geo.era_interim.cache Namespace Reference

Namespaces

- [data_fetcher](#)

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

Classes

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

5.14 skdaccess.geo.gldas Namespace Reference

Namespaces

- [data_fetcher](#)

5.15 skdaccess.geo.gldas.data_fetcher Namespace Reference

Classes

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

5.16 skdaccess.geo.grace Namespace Reference

Namespaces

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

5.17 skdaccess.geo.grace.data_fetcher Namespace Reference

Classes

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

5.18 skdaccess.geo.grace.mascon Namespace Reference

Namespaces

- [cache](#)

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

Namespaces

- [data_fetcher](#)

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

Classes

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

5.21 skdaccess.geo.groundwater Namespace Reference

Namespaces

- [data_fetcher](#)

5.22 skdaccess.geo.groundwater.data_fetcher Namespace Reference

Classes

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

5.23 skdaccess.geo.imsdnhs Namespace Reference

Namespaces

- [data_fetcher](#)

5.24 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.

5.25 skdaccess.geo.magnetometer Namespace Reference

Namespaces

- [data_fetcher](#)

5.26 skdaccess.geo.magnetometer.data_fetcher Namespace Reference

Classes

- class [DataFetcher](#)

Data fetcher for USGS geomagnetic observatories.

5.27 skdaccess.geo.mahali Namespace Reference

Namespaces

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

5.28 skdaccess.geo.mahali.rinex Namespace Reference

Namespaces

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

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

Classes

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

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

Classes

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

5.31 skdaccess.geo.mahali.tec Namespace Reference

Namespaces

- [data_fetcher](#)

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

Classes

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

5.33 skdaccess.geo.mahali.temperature Namespace Reference

Namespaces

- [data_fetcher](#)

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

Classes

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

5.35 skdaccess.geo.modis Namespace Reference

Namespaces

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

5.36 skdaccess.geo.modis.cache Namespace Reference

Namespaces

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

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

Namespaces

- [data_fetcher](#)

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

Classes

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

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

Namespaces

- [data_fetcher](#)

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

Classes

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

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

Classes

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

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

Namespaces

- [data_fetcher](#)

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

Classes

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

5.44 skdaccess.geo.modis.stream Namespace Reference

Namespaces

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

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

Namespaces

- [data_fetcher](#)

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

Classes

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

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

Namespaces

- [data_fetcher](#)

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

Classes

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

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

Classes

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

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

Namespaces

- [data_fetcher](#)

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

Classes

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

5.52 skdaccess.geo.ngl_gps Namespace Reference

Namespaces

- [data_fetcher](#)

5.53 skdaccess.geo.ngl_gps.data_fetcher Namespace Reference

Classes

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

5.54 skdaccess.geo.pbo Namespace Reference

Namespaces

- [data_fetcher](#)

5.55 skdaccess.geo.pbo.data_fetcher Namespace Reference

Classes

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

5.56 skdaccess.geo.sentinel_1 Namespace Reference

Namespaces

- [cache](#)

5.57 skdaccess.geo.sentinel_1.cache Namespace Reference

Namespaces

- [data_fetcher](#)

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

Classes

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

5.59 skdaccess.geo.srtm Namespace Reference

Namespaces

- [cache](#)

5.60 skdaccess.geo.srtm.cache Namespace Reference

Namespaces

- [data_fetcher](#)

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

Classes

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

5.62 skdaccess.geo.uavsar Namespace Reference

Namespaces

- [cache](#)

5.63 skdaccess.geo.uavsar.cache Namespace Reference

Namespaces

- [data_fetcher](#)

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

Classes

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

5.65 skdaccess.geo.wyoming_sounding Namespace Reference

Namespaces

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

5.66 skdaccess.geo.wyoming_sounding.cache Namespace Reference

Namespaces

- [data_fetcher](#)

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

Classes

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

5.68 skdaccess.geo.wyoming_sounding.stream Namespace Reference

Namespaces

- [data_fetcher](#)

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

Classes

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

5.70 skdaccess.planetary Namespace Reference

Namespaces

- [ode](#)

5.71 skdaccess.planetary.ode Namespace Reference

Namespaces

- [cache](#)

5.72 skdaccess.planetary.ode.cache Namespace Reference

Namespaces

- [data_fetcher](#)

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

Classes

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

5.74 skdaccess.solar Namespace Reference

Namespaces

- [sdo](#)

5.75 skdaccess.solar.sdo Namespace Reference

Namespaces

- [data_fetcher](#)

5.76 skdaccess.solar.sdo.data_fetcher Namespace Reference

Classes

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

5.77 skdaccess.utilities Namespace Reference

Namespaces

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

5.78 skdaccess.utilities.file_browser Namespace Reference

Classes

- class [FileBrowser](#)

5.79 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 [getStartDate](#) (in_data)

5.79.1 Function Documentation

5.79.1.1 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

5.79.1.2 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.

5.79.1.3 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

5.79.1.4 getStartEndDate()

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

5.79.1.5 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

5.80 skdaccess.utilities.gw_util Namespace Reference

Functions

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

5.80.1 Function Documentation

5.80.1.1 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

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

5.81 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](#)

5.81.1 Function Documentation

5.81.1.1 [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

5.81.1.2 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 )
```

5.81.1.3 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

5.81.1.4 SplineGeolocation()

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

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

5.81.2 Variable Documentation

5.81.2.1 lat_spline

`skdaccess.utilities.image_util.lat_spline`

5.81.2.2 lon_spline

`skdaccess.utilities.image_util.lon_spline`

5.81.2.3 x_offset

`skdaccess.utilities.image_util.x_offset`

5.81.2.4 x_spline

`skdaccess.utilities.image_util.x_spline`

5.81.2.5 y_offset

`skdaccess.utilities.image_util.y_offset`

5.81.2.6 y_spline

`skdaccess.utilities.image_util.y_spline`

5.82 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 by the median for the quarter.

5.82.1 Function Documentation

5.82.1.1 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 by the median for the quarter.

Parameters

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

5.83 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')

5.83.1 Function Documentation

5.83.1.1 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

5.83.1.2 `parseIonoFile()`

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

5.84 `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.

5.84.1 Function Documentation

5.84.1.1 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

5.84.1.2 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

5.84.1.3 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

5.84.1.4 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

5.84.1.5 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

5.84.1.6 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

5.84.1.7 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)

5.84.1.8 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')

5.84.1.9 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

5.85 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.

5.85.1 Function Documentation

5.85.1.1 `correct_CRISM_label()`

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

5.85.1.2 `correct_file_name_case_in_label()`

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

5.85.1.3 `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

5.85.1.4 `get_files_urls()`

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

5.85.1.5 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 )
```

5.85.1.6 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

5.85.1.7 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

5.85.1.8 `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'
<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

5.85.1.9 query_yes_no()

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

5.86 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.

5.86.1 Function Documentation

5.86.1.1 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 containing two tuples, *lat_range* and *lon_range*

5.86.1.2 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 <i>geo_point</i>
<i>data</i>	Stabilized (or unstabilized) data generated from the data fetcher or out of <i>stab_sys</i>
<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

5.86.1.3 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

5.86.1.4 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

5.86.1.5 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

5.86.1.6 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

5.86.1.7 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

5.87 skdaccess.utilities.sentinel_1_util Namespace Reference

Functions

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

5.87.1 Function Documentation

5.87.1.1 parseSatelliteData()

```

def skdaccess.utilities.sentinel_1_util.parseSatelliteData (
    in_satellite_file )

```

Parse Sentinel satellite data.

Parameters

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

Returns

DataFrame of orbit information

5.88 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.

5.88.1 Function Documentation

5.88.1.1 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

5.89 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.

5.89.1 Function Documentation

5.89.1.1 [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 latitude
<i>lat_end</i>	Ending latitude
<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

5.89.1.2 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)

5.89.1.3 merge_srtm_tiles()

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

5.90 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)

5.90.1 Function Documentation

5.90.1.1 convertToStr()

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

5.90.1.2 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

5.90.1.3 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

5.91 skdaccess.utilities.uavsar_util Namespace Reference

Functions

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

5.91.1 Function Documentation

5.91.1.1 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

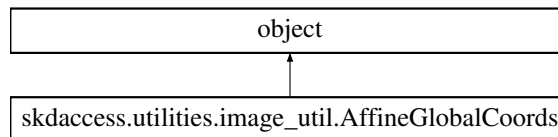
Chapter 6

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

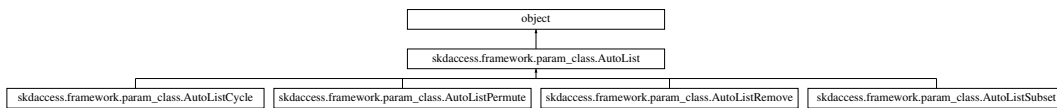
The documentation for this class was generated from the following file:

- [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
```

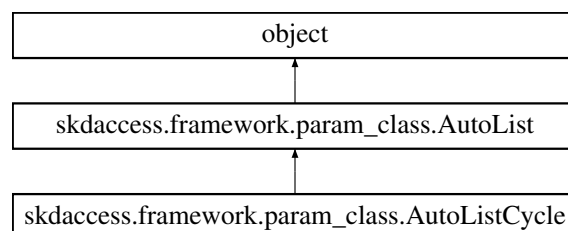
The documentation for this class was generated from the following file:

- 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 parmaters 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`

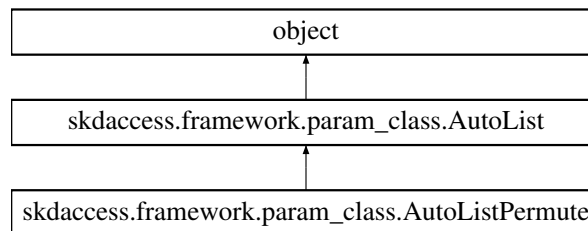
The documentation for this class was generated from the following file:

- [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]

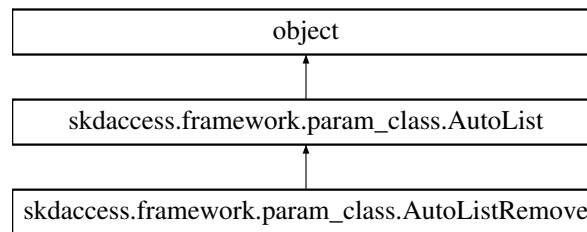
The documentation for this class was generated from the following file:

- [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
```

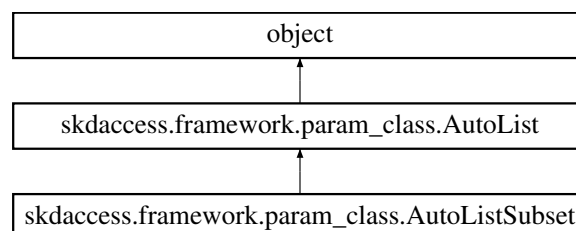
The documentation for this class was generated from the following file:

- 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
```

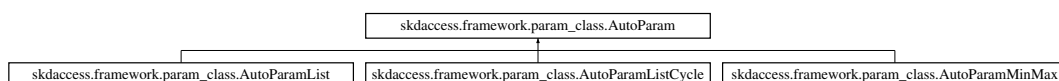
The documentation for this class was generated from the following file:

- [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`

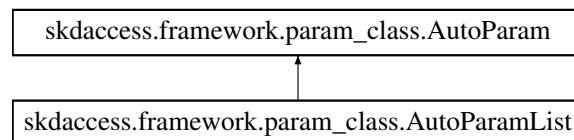
The documentation for this class was generated from the following file:

- [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`

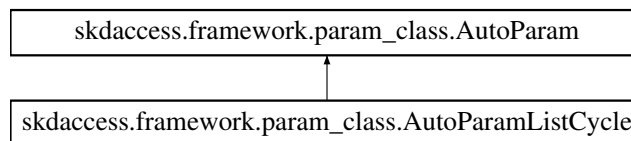
The documentation for this class was generated from the following file:

- [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
```

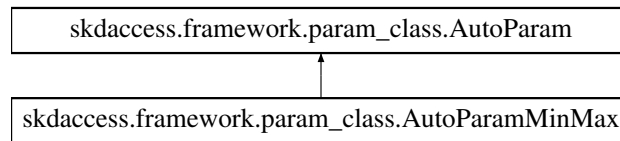
The documentation for this class was generated from the following file:

- 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
```

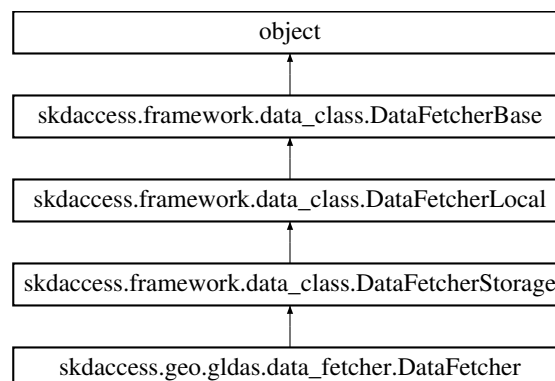
The documentation for this class was generated from the following file:

- [framework/param_class.py](#)

6.11 skdaccess.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 `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`

6.11.1 Detailed Description

Data Fetcher for GLDAS data.

6.11.2 Constructor & Destructor Documentation

6.11.2.1 `__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[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)

6.11.3 Member Function Documentation

6.11.3.1 `__str__()`

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

String representation of data fetcher.

Returns

String listing the name and geopoint of data fetcher

6.11.3.2 `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

6.11.3.3 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.11.3.4 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

6.11.3.5 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.11.3.6 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

6.11.3.7 output()

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

Create data wrapper of GLDAS data for specified geopoint.

Returns

GLDAS Data Wrapper

6.11.3.8 perturb()

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

Perturb parameters.

6.11.3.9 reset()

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

Set all parameters to initial value.

6.11.3.10 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

6.11.3.11 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

6.11.3.12 writeConfig()

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

Write config to disk.

Parameters

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

6.11.4 Member Data Documentation

6.11.4.1 ap_paramList

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

6.11.4.2 `end_date`

`skdaccess.geo.gldas.DataFetcher.end_date`

6.11.4.3 `resample`

`skdaccess.geo.gldas.DataFetcher.resample`

6.11.4.4 `start_date`

`skdaccess.geo.gldas.DataFetcher.start_date`

6.11.4.5 `verbose`

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

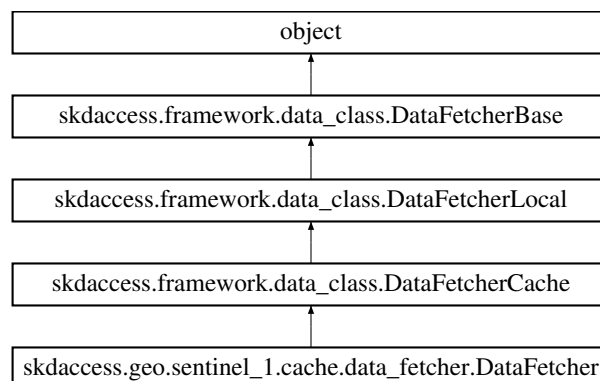
The documentation for this class was generated from the following file:

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

6.12 `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 `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

6.12.1 Detailed Description

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

6.12.2 Constructor & Destructor Documentation

6.12.2.1 `__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

6.12.3 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 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

6.12.3.6 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

6.12.3.7 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.12.3.8 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

6.12.3.9 output()

```
def skdaccess.geo.sentinel_1.cache.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

Sentinel SLC data in a data wrapper

6.12.3.10 perturb()

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

Perturb parameters.

6.12.3.11 reset()

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

Set all parameters to initial value.

6.12.3.12 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

6.12.3.13 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

6.12.3.14 writeConfig()

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

Write config to disk.

Parameters

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

6.12.4 Member Data Documentation**6.12.4.1 ap_paramList**

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

6.12.4.2 local_paths

```
skdaccess.geo.sentinel_1.cache.DataFetcher.local_paths
```

6.12.4.3 password

```
skdaccess.geo.sentinel_1.cache.DataFetcher.password
```

6.12.4.4 polarization

`skdaccess.geo.sentinel_1.cache.DataFetcher.polarization`

6.12.4.5 satellite_url_list

`skdaccess.geo.sentinel_1.cache.DataFetcher.satellite_url_list`

6.12.4.6 swath

`skdaccess.geo.sentinel_1.cache.DataFetcher.swath`

6.12.4.7 url_list

`skdaccess.geo.sentinel_1.cache.DataFetcher.url_list`

6.12.4.8 username

`skdaccess.geo.sentinel_1.cache.DataFetcher.username`

6.12.4.9 verbose

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

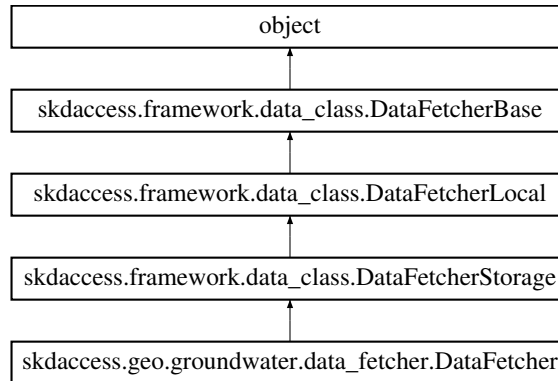
The documentation for this class was generated from the following file:

- `geo/sentinel_1/cache/data_fetcher.py`

6.13 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 `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](#)

6.13.1 Detailed Description

Generates Data Wrappers of groundwater measurements taken in the US.

6.13.2 Constructor & Destructor Documentation

6.13.2.1 `__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

6.13.3 Member Function Documentation

6.13.3.1 `__str__()`

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

String representation of data fetcher.

Returns

string describing data fetcher

6.13.3.2 `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

6.13.3.3 `getConfig()`

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.13.3.4 `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

6.13.3.5 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.13.3.6 getStationMetadata()

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

Retrieve metadata on groundwater wells.

Returns

pandas dataframe with groundwater well information

6.13.3.7 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

6.13.3.8 output()

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

Fetch Groundwater Data Wrapper.

Returns

Groundwater Data Wrapper

6.13.3.9 perturb()

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

Perturb parameters.

6.13.3.10 reset()

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

Set all parameters to initial value.

6.13.3.11 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

6.13.3.12 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

6.13.3.13 writeConfig()

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

Write config to disk.

Parameters

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

6.13.4 Member Data Documentation**6.13.4.1 ap_paramList**

```
skdaccess.geo.groundwater.DataFetcher.ap_paramList
```

6.13.4.2 cutoff

```
skdaccess.geo.groundwater.DataFetcher.cutoff
```

6.13.4.3 end_date

`skdaccess.geo.groundwater.DataFetcher.end_date`

6.13.4.4 start_date

`skdaccess.geo.groundwater.DataFetcher.start_date`

6.13.4.5 verbose

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

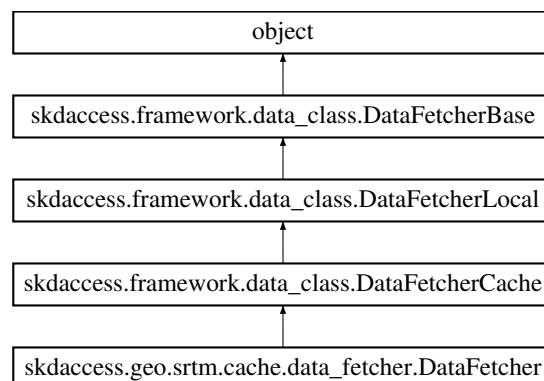
The documentation for this class was generated from the following file:

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

6.14 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 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`

6.14.1 Detailed Description

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

6.14.2 Constructor & Destructor Documentation

6.14.2.1 __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

6.14.3 Member Function Documentation

6.14.3.1 __str__()

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

Generate string description.

6.14.3.2 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

6.14.3.3 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

6.14.3.4 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.14.3.5 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

6.14.3.6 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

6.14.3.7 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.14.3.8 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

6.14.3.9 output()

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

Generate SRTM data wrapper.

Returns

SRTM Image Wrapper

6.14.3.10 perturb()

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

Perturb parameters.

6.14.3.11 reset()

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

Set all parameters to initial value.

6.14.3.12 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

6.14.3.13 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

6.14.3.14 writeConfig()

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

Write config to disk.

Parameters

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

6.14.4 Member Data Documentation**6.14.4.1 ap_paramList**

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

6.14.4.2 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

6.14.4.3 lat_tile_end

skdaccess.geo.srtm.cache.DataFetcher.lat_tile_end

6.14.4.4 lat_tile_start

skdaccess.geo.srtm.cache.DataFetcher.lat_tile_start

6.14.4.5 lon_tile_end

skdaccess.geo.srtm.cache.DataFetcher.lon_tile_end

6.14.4.6 lon_tile_start

skdaccess.geo.srtm.cache.DataFetcher.lon_tile_start

6.14.4.7 mask_water

skdaccess.geo.srtm.cache.DataFetcher.mask_water

6.14.4.8 password

skdaccess.geo.srtm.cache.DataFetcher.password

6.14.4.9 store_geolocation_grids

skdaccess.geo.srtm.cache.DataFetcher.store_geolocation_grids

6.14.4.10 username

`skdaccess.geo.srtm.cache.DataFetcher.username`

6.14.4.11 verbose

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

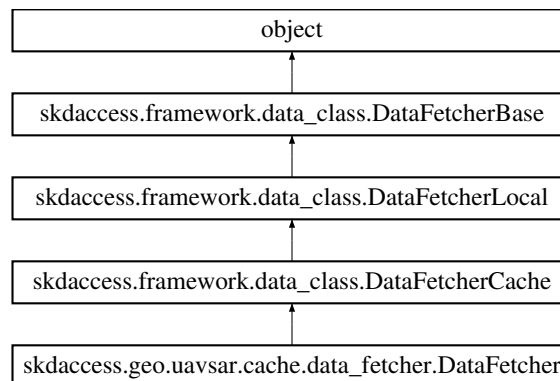
The documentation for this class was generated from the following file:

- [geo/srtm/cache/data_fetcher.py](#)

6.15 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 [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](#)

6.15.1 Detailed Description

Data Fetcher for UAVSAR data.

6.15.2 Constructor & Destructor Documentation

6.15.2.1 `__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

6.15.3 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
<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 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.15.3.5 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

6.15.3.6 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

6.15.3.7 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.15.3.8 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

6.15.3.9 output()

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

Output data as a data wrapper.

Returns

Imagewriter of data

6.15.3.10 perturb()

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

Perturb parameters.

6.15.3.11 reset()

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

Set all parameters to initial value.

6.15.3.12 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

6.15.3.13 `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

6.15.3.14 `writeConfig()`

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

Write config to disk.

Parameters

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

6.15.4 Member Data Documentation

6.15.4.1 `ap_paramList`

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

6.15.4.2 llh_url

`skdaccess.geo.uavsar.cache.DataFetcher.llh_url`

6.15.4.3 memmap

`skdaccess.geo.uavsar.cache.DataFetcher.memmap`

6.15.4.4 metadata_url_list

`skdaccess.geo.uavsar.cache.DataFetcher.metadata_url_list`

6.15.4.5 slc_url_list

`skdaccess.geo.uavsar.cache.DataFetcher.slc_url_list`

6.15.4.6 verbose

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

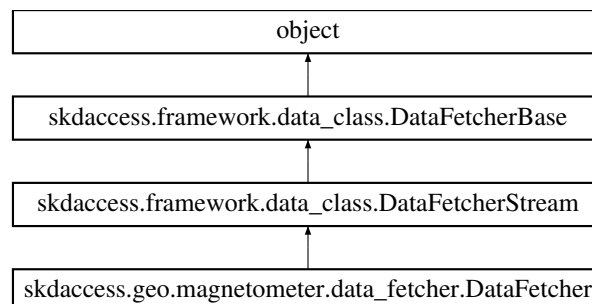
The documentation for this class was generated from the following file:

- [geo/uavsar/cache/data_fetcher.py](#)

6.16 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 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`

6.16.1 Detailed Description

Data fetcher for USGS geomagnetic observatories.

6.16.2 Constructor & Destructor Documentation

6.16.2.1 `__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

6.16.3 Member Function Documentation

6.16.3.1 `__str__()`

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

Generate string description.

6.16.3.2 `getConfig()`

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.16.3.3 `getDataMetadata()`

```
def skdaccess.geo.magnetometer.DataFetcher.getDataMetadata ( )
```

Get data metadata.

Returns

Pandas dataframe containing station latitude and longitude coordinates

6.16.3.4 `getMetadata()`

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.16.3.5 `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

6.16.3.6 `output()`

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

Generate data wrapper for USGS geomagnetic data.

Returns

geomagnetic data wrapper

6.16.3.7 perturb()

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

Perturb parameters.

6.16.3.8 reset()

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

Set all parameters to initial value.

6.16.3.9 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

6.16.3.10 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

6.16.3.11 writeConfig()

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

Write config to disk.

Parameters

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

6.16.4 Member Data Documentation**6.16.4.1 ap_paramList**

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

6.16.4.2 channels

```
skdaccess.geo.magnetometer.DataFetcher.channels
```

6.16.4.3 data_type

```
skdaccess.geo.magnetometer.DataFetcher.data_type
```

6.16.4.4 end_time

`skdaccess.geo.magnetometer.DataFetcher.end_time`

6.16.4.5 interval

`skdaccess.geo.magnetometer.DataFetcher.interval`

6.16.4.6 start_time

`skdaccess.geo.magnetometer.DataFetcher.start_time`

6.16.4.7 verbose

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

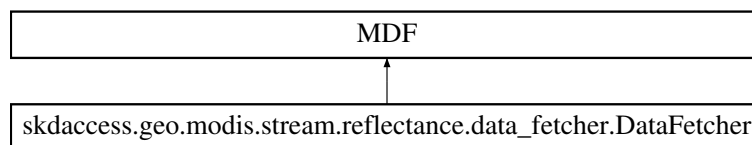
The documentation for this class was generated from the following file:

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

6.17 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.

6.17.1 Detailed Description

Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

6.17.2 Constructor & Destructor Documentation

6.17.2.1 `__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

<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

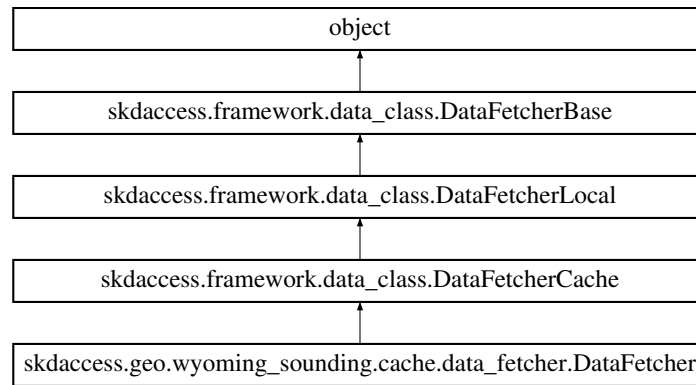
The documentation for this class was generated from the following file:

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

6.18 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 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 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](#)

6.18.1 Detailed Description

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

6.18.2 Constructor & Destructor Documentation

6.18.2.1 `__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)

6.18.3 Member Function Documentation

6.18.3.1 __str__()

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

Generate string description.

6.18.3.2 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

6.18.3.3 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

6.18.3.4 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.18.3.5 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

6.18.3.6 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

6.18.3.7 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.18.3.8 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

6.18.3.9 output()

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

Generate data wrapper.

Returns

Wyoming sounding data in a data wrapper

6.18.3.10 perturb()

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

Perturb parameters.

6.18.3.11 reset()

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

Set all parameters to initial value.

6.18.3.12 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

6.18.3.13 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

6.18.3.14 writeConfig()

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

Write config to disk.

Parameters

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

6.18.4 Member Data Documentation

6.18.4.1 ap_paramList

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

6.18.4.2 day_end

```
skdaccess.geo.wyoming_sounding.cache.DataFetcher.day_end
```

6.18.4.3 day_start

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.day_start`

6.18.4.4 end_hour

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.end_hour`

6.18.4.5 month_list

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.month_list`

6.18.4.6 start_hour

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.start_hour`

6.18.4.7 station_number

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.station_number`

6.18.4.8 verbose

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

6.18.4.9 year_list

`skdaccess.geo.wyoming_sounding.cache.DataFetcher.year_list`

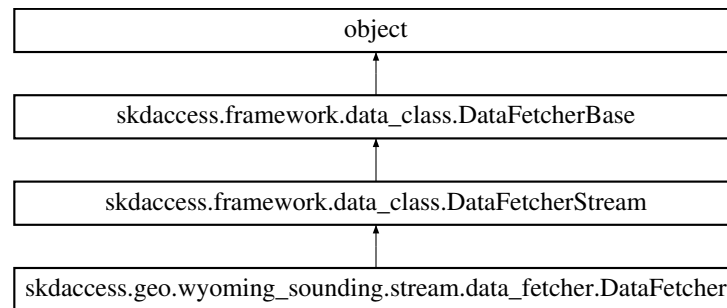
The documentation for this class was generated from the following file:

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

6.19 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 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 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.19.3.4 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

6.19.3.5 output() [1/2]

```
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

6.19.3.6 output() [2/2]

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

Output data wrapper.

Returns

Datawrapper

6.19.3.7 perturb()

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

Perturb parameters.

6.19.3.8 reset()

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

Set all parameters to initial value.

6.19.3.9 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

6.19.3.10 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

6.19.3.11 writeConfig()

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

Write config to disk.

Parameters

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

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

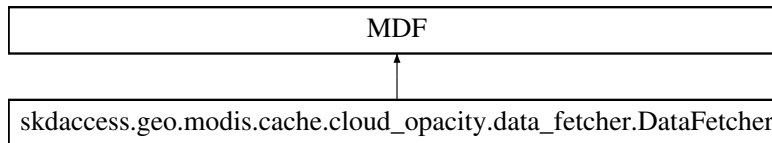
The documentation for this class was generated from the following file:

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

6.20 skdaccess.geo.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.

6.20.1 Detailed Description

Data Fetcher for MODIS Cloud Opacity.

6.20.2 Constructor & Destructor Documentation

6.20.2.1 __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

<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)

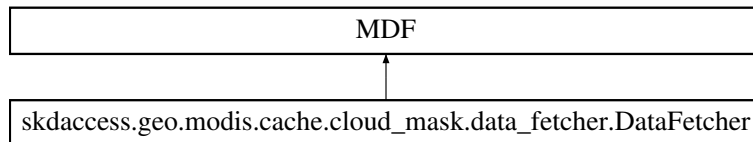
The documentation for this class was generated from the following file:

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

6.21 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.

6.21.1 Detailed Description

Data Fetcher for MODIS Cloud Mask.

6.21.2 Constructor & Destructor Documentation

6.21.2.1 `__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

<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)

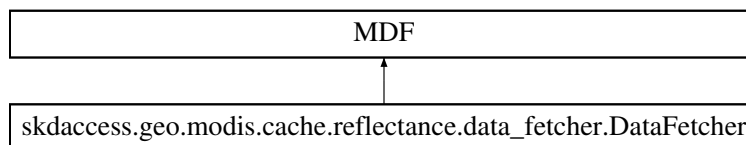
The documentation for this class was generated from the following file:

- `geo/modis/cache/cloud_mask/data_fetcher.py`

6.22 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.

6.22.1 Detailed Description

Data fetcher for the modis surface reflectance product ('09', 1 km resolution)

6.22.2 Constructor & Destructor Documentation

6.22.2.1 `__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

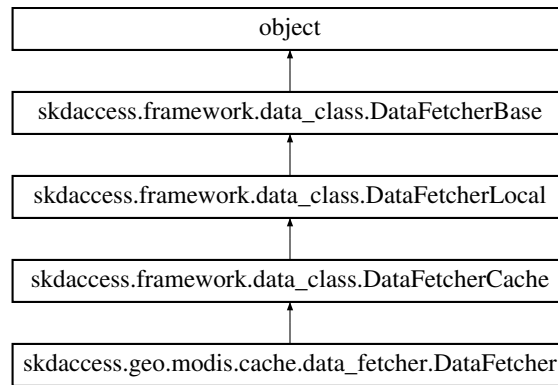
The documentation for this class was generated from the following file:

- [geo/modis/cache/reflectance/data_fetcher.py](#)

6.23 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 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 `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`

6.23.1 Detailed Description

Data Fetcher for MODIS data.

6.23.2 Constructor & Destructor Documentation

6.23.2.1 `__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
<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

6.23.3 Member Function Documentation

6.23.3.1 `__str__()`

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

Generate string description.

6.23.3.2 `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
---------------------------	---------------------------

6.23.3.3 `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 <code>use_requests=True</code>)
<i>cookiejar</i>	The cookiejar that stores credentials (unused when <code>use_requests=True</code>)
<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

6.23.3.4 `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

6.23.3.5 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

6.23.3.6 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.23.3.7 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

6.23.3.8 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

6.23.3.9 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.23.3.10 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

6.23.3.11 output()

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

Generate data wrapper.

Returns

data wrapper of MODIS data

6.23.3.12 perturb()

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

Perturb parameters.

6.23.3.13 reset()

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

Set all parameters to initial value.

6.23.3.14 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

6.23.3.15 `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

6.23.3.16 `writeConfig()`

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

Write config to disk.

Parameters

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

6.23.4 Member Data Documentation

6.23.4.1 `ap_paramList`

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

6.23.4.2 daynightboth

skdaccess.geo.modis.cache.DataFetcher.daynightboth

6.23.4.3 end_date

skdaccess.geo.modis.cache.DataFetcher.end_date

6.23.4.4 grid

skdaccess.geo.modis.cache.DataFetcher.grid

6.23.4.5 grid_fill

skdaccess.geo.modis.cache.DataFetcher.grid_fill

6.23.4.6 modis_id

skdaccess.geo.modis.cache.DataFetcher.modis_id

6.23.4.7 modis_identifier

skdaccess.geo.modis.cache.DataFetcher.modis_identifier

6.23.4.8 modis_platform

skdaccess.geo.modis.cache.DataFetcher.modis_platform

6.23.4.9 start_date

```
skdaccess.geo.modis.cache.DataFetcher.start_date
```

6.23.4.10 use_long_name

```
skdaccess.geo.modis.cache.DataFetcher.use_long_name
```

6.23.4.11 variable_list

```
skdaccess.geo.modis.cache.DataFetcher.variable_list
```

6.23.4.12 verbose

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

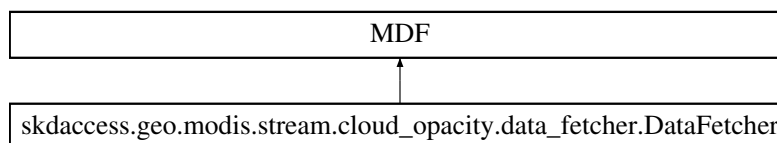
The documentation for this class was generated from the following file:

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

6.24 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.

6.24.1 Detailed Description

Data Fetcher for MODIS Cloud Opacity.

6.24.2 Constructor & Destructor Documentation

6.24.2.1 __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)

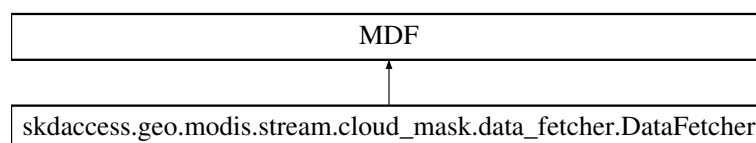
The documentation for this class was generated from the following file:

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

6.25 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.

6.25.1 Detailed Description

Data Fetcher for MODIS Cloud Mask.

6.25.2 Constructor & Destructor Documentation

6.25.2.1 `__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

<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)

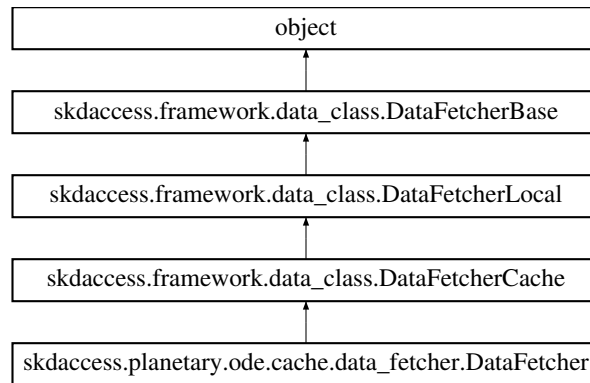
The documentation for this class was generated from the following file:

- `geo/modis/stream/cloud_mask/data_fetcher.py`

6.26 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')
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 `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](#)

6.26.1 Detailed Description

Data Fetcher from the Orbital Data Explorer (ODE)

6.26.2 Constructor & Destructor Documentation

6.26.2.1 `__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 )
```

6.26.3 Member Function Documentation

6.26.3.1 `__str__()`

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

Generate string description.

6.26.3.2 `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

6.26.3.3 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

6.26.3.4 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.26.3.5 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

6.26.3.6 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

6.26.3.7 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.26.3.8 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

6.26.3.9 output()

```
def skdaccess.planetary.ode.cache.DataFetcher.output (
    self )
```

Generate data wrapper from ODE data.

6.26.3.10 perturb()

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

Perturb parameters.

6.26.3.11 reset()

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

Set all parameters to initial value.

6.26.3.12 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

6.26.3.13 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

6.26.3.14 writeConfig()

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

Write config to disk.

Parameters

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

6.26.4 Member Data Documentation

6.26.4.1 ap_paramList

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

6.26.4.2 eastern_lon

```
skdaccess.planetary.ode.cache.DataFetcher.eastern_lon
```

6.26.4.3 file_name

`skdaccess.planetary.ode.cache.DataFetcher.file_name`

6.26.4.4 instrument

`skdaccess.planetary.ode.cache.DataFetcher.instrument`

6.26.4.5 max_lat

`skdaccess.planetary.ode.cache.DataFetcher.max_lat`

6.26.4.6 max_ob_time

`skdaccess.planetary.ode.cache.DataFetcher.max_ob_time`

6.26.4.7 min_lat

`skdaccess.planetary.ode.cache.DataFetcher.min_lat`

6.26.4.8 min_ob_time

`skdaccess.planetary.ode.cache.DataFetcher.min_ob_time`

6.26.4.9 mission

`skdaccess.planetary.ode.cache.DataFetcher.mission`

6.26.4.10 number_product_limit

`skdaccess.planetary.ode.cache.DataFetcher.number_product_limit`

6.26.4.11 product_id

`skdaccess.planetary.ode.cache.DataFetcher.product_id`

6.26.4.12 product_type

`skdaccess.planetary.ode.cache.DataFetcher.product_type`

6.26.4.13 remove_ndv

`skdaccess.planetary.ode.cache.DataFetcher.remove_ndv`

6.26.4.14 result_offset_number

`skdaccess.planetary.ode.cache.DataFetcher.result_offset_number`

6.26.4.15 target

`skdaccess.planetary.ode.cache.DataFetcher.target`

6.26.4.16 verbose

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

6.26.4.17 western_lon

`skdaccess.planetary.ode.cache.DataFetcher.western_lon`

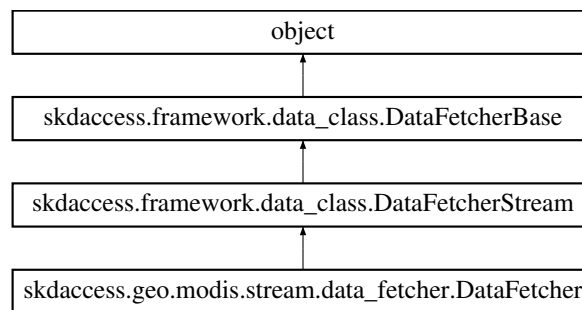
The documentation for this class was generated from the following file:

- [planetary/ode/cache/data_fetcher.py](#)

6.27 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 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](#)

6.27.1 Detailed Description

Data Fetcher for MODIS data.

6.27.2 Constructor & Destructor Documentation

6.27.2.1 `__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

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

6.27.3 Member Function Documentation**6.27.3.1 __str__()**

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

Generate string description.

6.27.3.2 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.27.3.3 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.27.3.4 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

6.27.3.5 output()

```
def skdaccess.geo.modis.stream.DataFetcher.output (
    self )
```

Generate data wrapper.

Returns

data wrapper of MODIS data

6.27.3.6 perturb()

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

Perturb parameters.

6.27.3.7 reset()

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

Set all parameters to initial value.

6.27.3.8 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

6.27.3.9 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

6.27.3.10 writeConfig()

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

Write config to disk.

Parameters

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

6.27.4 Member Data Documentation

6.27.4.1 ap_paramList

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

6.27.4.2 daynightboth

skdaccess.geo.modis.stream.DataFetcher.daynightboth

6.27.4.3 end_date

skdaccess.geo.modis.stream.DataFetcher.end_date

6.27.4.4 grid

skdaccess.geo.modis.stream.DataFetcher.grid

6.27.4.5 grid_fill

skdaccess.geo.modis.stream.DataFetcher.grid_fill

6.27.4.6 modis_id

skdaccess.geo.modis.stream.DataFetcher.modis_id

6.27.4.7 modis_identifier

skdaccess.geo.modis.stream.DataFetcher.modis_identifier

6.27.4.8 modis_platform

`skdaccess.geo.modis.stream.DataFetcher.modis_platform`

6.27.4.9 start_date

`skdaccess.geo.modis.stream.DataFetcher.start_date`

6.27.4.10 use_long_name

`skdaccess.geo.modis.stream.DataFetcher.use_long_name`

6.27.4.11 variable_list

`skdaccess.geo.modis.stream.DataFetcher.variable_list`

6.27.4.12 verbose

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

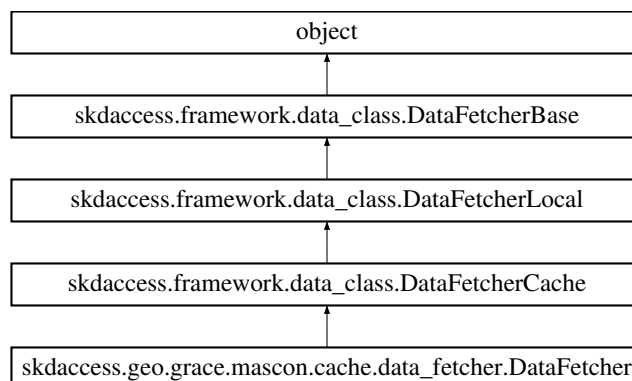
The documentation for this class was generated from the following file:

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

6.28 skdaccess.geo.grace.mascon.cache.DataFetcher Class Reference

Data Fetcher for GRACE mascon data.

Inheritance diagram for `skdaccess.geo.grace.mascon.cache.data_fetcher.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 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 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`

6.28.1 Detailed Description

Data Fetcher for GRACE mascon data.

6.28.2 Constructor & Destructor Documentation

6.28.2.1 `__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</i> [<i>geo_point</i>]	AutoList of geographic location tuples (lat,lon)
<i>start_date</i>	Beginning date
<i>end_date</i>	Ending date

6.28.3 Member Function Documentation

6.28.3.1 `__str__()`

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

Generate string description.

6.28.3.2 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

6.28.3.3 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

6.28.3.4 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

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 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

6.28.3.7 getMasconPlacement()

```
def skdaccess.geo.grace.mascon.cache.DataFetcher.getMasconPlacement (
    self )
```

Retrieve mascon placement data.

Returns

Mascon data, Mascon metadata

6.28.3.8 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.28.3.9 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

6.28.3.10 output()

```
def skdaccess.geo.grace.mascon.cache.DataFetcher.output (
    self )
```

Create a datawrapper containing GRACE mascon data.

Returns

Table Datawrapper containing Mascon GRACE data

6.28.3.11 perturb()

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

Perturb parameters.

6.28.3.12 reset()

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

Set all parameters to initial value.

6.28.3.13 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

6.28.3.14 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

6.28.3.15 writeConfig()

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

Write config to disk.

Parameters

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

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.grace.mascon.cache.DataFetcher.end_date
```

6.28.4.3 mascon_placement_url

`skdaccess.geo.grace.mascon.cache.DataFetcher.mascon_placement_url`

6.28.4.4 mascon_url

`skdaccess.geo.grace.mascon.cache.DataFetcher.mascon_url`

6.28.4.5 scale_factor_url

`skdaccess.geo.grace.mascon.cache.DataFetcher.scale_factor_url`

6.28.4.6 start_date

`skdaccess.geo.grace.mascon.cache.DataFetcher.start_date`

6.28.4.7 verbose

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

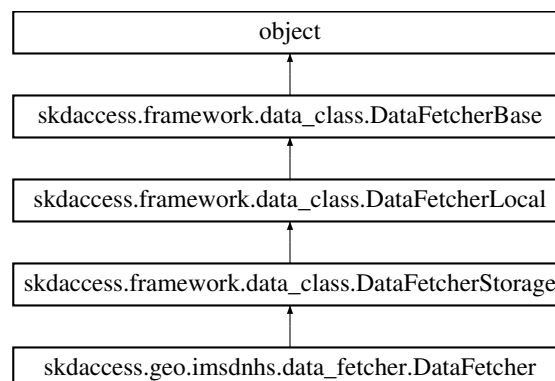
The documentation for this class was generated from the following file:

- [geo/grace/mascon/cache/data_fetcher.py](#)

6.29 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 `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`

6.29.1 Detailed Description

Fetches data for the Interactive Multisensor Snow and Ice Mapping System Daily Northern Hemisphere Snow and Ice Analysis.

6.29.2 Constructor & Destructor Documentation

6.29.2.1 `__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

6.29.3 Member Function Documentation

6.29.3.1 `__str__()`

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

Generate string description.

6.29.3.2 `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

6.29.3.3 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.29.3.4 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

6.29.3.5 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.29.3.6 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

6.29.3.7 output()

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

Fetch snow coverage data for coordinates.

Returns

Data wrapper for snow coverage

6.29.3.8 perturb()

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

Perturb parameters.

6.29.3.9 reset()

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

Set all parameters to initial value.

6.29.3.10 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

6.29.3.11 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

6.29.3.12 writeConfig()

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

Write config to disk.

Parameters

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

6.29.4 Member Data Documentation

6.29.4.1 ap_paramList

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

6.29.4.2 coordinate_dict

`skdaccess.geo.imsdnhs.DataFetcher.coordinate_dict`

6.29.4.3 end_date

`skdaccess.geo.imsdnhs.DataFetcher.end_date`

6.29.4.4 start_date

`skdaccess.geo.imsdnhs.DataFetcher.start_date`

6.29.4.5 verbose

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

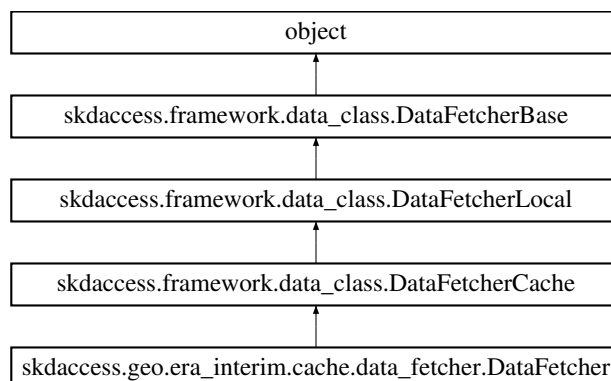
The documentation for this class was generated from the following file:

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

6.30 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 [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 [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](#)

6.30.1 Detailed Description

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

6.30.2 Constructor & Destructor Documentation

6.30.2.1 `__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

6.30.3 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 `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

6.30.3.6 `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

6.30.3.7 `getMetadata()`

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.30.3.8 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

6.30.3.9 output()

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

Generate data wrapper.

Returns

Era-I weather in a data wrapper

6.30.3.10 perturb()

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

Perturb parameters.

6.30.3.11 reset()

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

Set all parameters to initial value.

6.30.3.12 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

6.30.3.13 `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

6.30.3.14 `writeConfig()`

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

Write config to disk.

Parameters

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

6.30.4 Member Data Documentation

6.30.4.1 `ap_paramList`

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

6.30.4.2 data_names

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

6.30.4.3 date_list

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

6.30.4.4 password

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

6.30.4.5 username

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

6.30.4.6 verbose

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

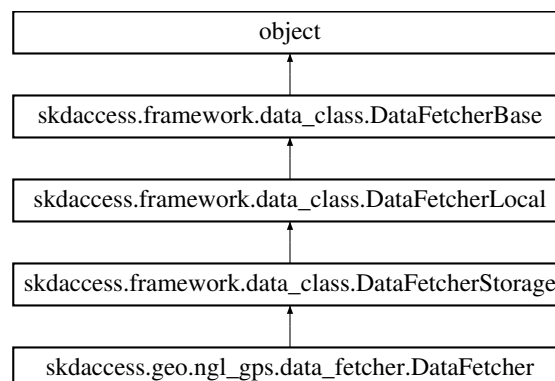
The documentation for this class was generated from the following file:

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

6.31 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')
Consctruct 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 `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`

6.31.1 Detailed Description

Data fetcher for GPS data from Nevada Geodetic Laboratory.

6.31.2 Constructor & Destructor Documentation

6.31.2.1 __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')

6.31.3 Member Function Documentation

6.31.3.1 __str__()

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

Generate string description.

6.31.3.2 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

6.31.3.3 getAntennaLogs()

```
def skdaccess.geo.ngl_gps.DataFetcher.getAntennaLogs ( )
```

Retrieve information about antenna changes.

Returns

dictionary of antenna changes

6.31.3.4 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.31.3.5 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

6.31.3.6 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.31.3.7 getStationMetadata()

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

Get station metadata.

Returns

data frame of station metadata

6.31.3.8 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

6.31.3.9 output()

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

Construct NGL GPS data wrapper.

Returns

NGL GPS data wrapper

6.31.3.10 perturb()

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

Perturb parameters.

6.31.3.11 reset()

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

Set all parameters to initial value.

6.31.3.12 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

6.31.3.13 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

6.31.3.14 writeConfig()

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

Write config to disk.

Parameters

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

6.31.4 Member Data Documentation

6.31.4.1 ap_paramList

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

6.31.4.2 data_type

```
skdaccess.geo.ngl_gps.DataFetcher.data_type
```

6.31.4.3 end_date

`skdaccess.geo.ngl_gps.DataFetcher.end_date`

6.31.4.4 lat_range

`skdaccess.geo.ngl_gps.DataFetcher.lat_range`

6.31.4.5 lon_range

`skdaccess.geo.ngl_gps.DataFetcher.lon_range`

6.31.4.6 mdyratio

`skdaccess.geo.ngl_gps.DataFetcher.mdyratio`

6.31.4.7 start_date

`skdaccess.geo.ngl_gps.DataFetcher.start_date`

6.31.4.8 verbose

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

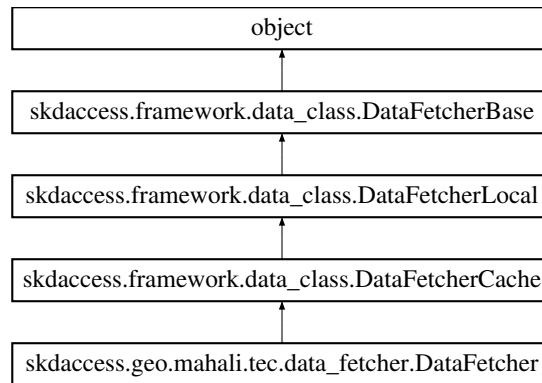
The documentation for this class was generated from the following file:

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

6.32 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 [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](#)

6.32.1 Detailed Description

Data Fetcher for Mahali Data.

6.32.2 Constructor & Destructor Documentation

6.32.2.1 `__init__()`

```
def skdaccess.geo.mahali.tec.DataFetcher.__init__ (
    self,
    ap_paramList = [],
    start_date = None,
    end_date = None )
```

Initialize Mahali 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)

6.32.3 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 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

6.32.3.6 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

6.32.3.7 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.32.3.8 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

6.32.3.9 output()

```
def skdaccess.geo.mahali.tec.DataFetcher.output (
    self )
```

Generate data wrapper for Mahali tec data.

Returns

Mahali data wrapper

6.32.3.10 perturb()

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

Perturb parameters.

6.32.3.11 reset()

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

Set all parameters to initial value.

6.32.3.12 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

6.32.3.13 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

6.32.3.14 writeConfig()

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

Write config to disk.

Parameters

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

6.32.4 Member Data Documentation**6.32.4.1 ap_paramList**

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

6.32.4.2 date_range

```
skdaccess.geo.mahali.tec.DataFetcher.date_range
```

6.32.4.3 end_date

```
skdaccess.geo.mahali.tec.DataFetcher.end_date
```

6.32.4.4 start_date

`skdaccess.geo.mahali.tec.DataFetcher.start_date`

6.32.4.5 verbose

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

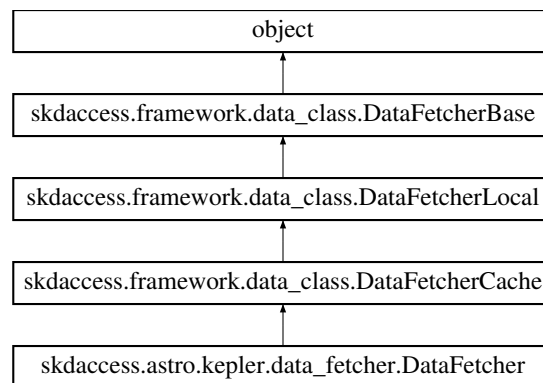
The documentation for this class was generated from the following file:

- [geo/mahali/tec/data_fetcher.py](#)

6.33 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 [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](#)

6.33.1 Detailed Description

Data Fetcher for Kepler light curve data.

6.33.2 Constructor & Destructor Documentation

6.33.2.1 `__init__()`

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

Initialize Kepler Data Fetcher.

Parameters

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

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 `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
---------------------------	--------------------

6.33.3.3 `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

6.33.3.4 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

6.33.3.5 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

6.33.3.6 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.33.3.7 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

6.33.3.8 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

6.33.3.9 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.33.3.10 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

6.33.3.11 output()

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

Output kepler data wrapper.

Returns

DataWrapper

6.33.3.12 perturb()

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

Perturb parameters.

6.33.3.13 reset()

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

Set all parameters to initial value.

6.33.3.14 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

6.33.3.15 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

6.33.3.16 writeConfig()

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

Write config to disk.

Parameters

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

6.33.4 Member Data Documentation**6.33.4.1 ap_paramList**

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

6.33.4.2 quarter_list

```
skdaccess.astro.kepler.DataFetcher.quarter_list
```

6.33.4.3 verbose

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

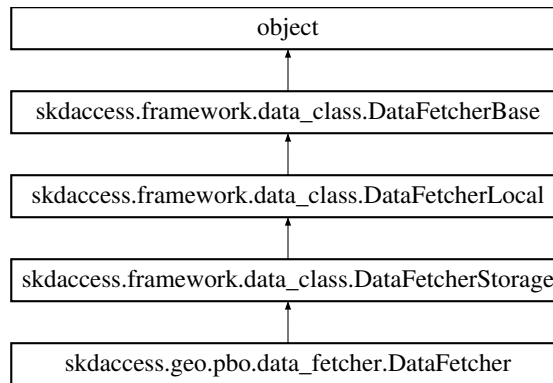
The documentation for this class was generated from the following file:

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

6.34 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 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](#)

6.34.1 Detailed Description

Data fetcher for PBO GPS data.

6.34.2 Constructor & Destructor Documentation

6.34.2.1 `__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)

6.34.3 Member Function Documentation

6.34.3.1 __str__()

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

print the parameter values

Returns

String representation of Data Fetcher

6.34.3.2 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

6.34.3.3 getAntennaLogs()

```
def skdaccess.geo.pbo.DataFetcher.getAntennaLogs ( )
```

Get antenna logs.

Returns

dictionary of data frames containing antenna logs

6.34.3.4 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.34.3.5 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

6.34.3.6 getInfo()

```
def skdaccess.geo.pbo.DataFetcher.getInfo (
    self )
```

Get information about the stations and geo_point.

Returns

tuple containing station list and geo_point

6.34.3.7 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.34.3.8 getStationMetadata()

```
def skdaccess.geo.pbo.DataFetcher.getStationMetadata (
    data_frame = False )
```

Read in the metadata and convert to dictionary.

Returns

dictionary of PBO metadata

6.34.3.9 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

6.34.3.10 output()

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

Generate PBO Data Wrapper.

Returns

PBO Data Wrapper

6.34.3.11 perturb()

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

Perturb parameters.

6.34.3.12 reset()

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

Set all parameters to initial value.

6.34.3.13 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

6.34.3.14 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
---------------------	---------------------------

6.34.3.15 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

6.34.3.16 writeConfig()

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

Write config to disk.

Parameters

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

6.34.4 Member Data Documentation

6.34.4.1 antenna_info

`skdaccess.geo.pbo.DataFetcher.antenna_info`

6.34.4.2 ap_paramList

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

6.34.4.3 default_columns

`skdaccess.geo.pbo.DataFetcher.default_columns`

6.34.4.4 default_error_columns

`skdaccess.geo.pbo.DataFetcher.default_error_columns`

6.34.4.5 index_date_only

`skdaccess.geo.pbo.DataFetcher.index_date_only`

6.34.4.6 meta_data

`skdaccess.geo.pbo.DataFetcher.meta_data`

6.34.4.7 station_list

`skdaccess.geo.pbo.DataFetcher.station_list`

6.34.4.8 use_progress_bar

`skdaccess.geo.pbo.DataFetcher.use_progress_bar`

6.34.4.9 verbose

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

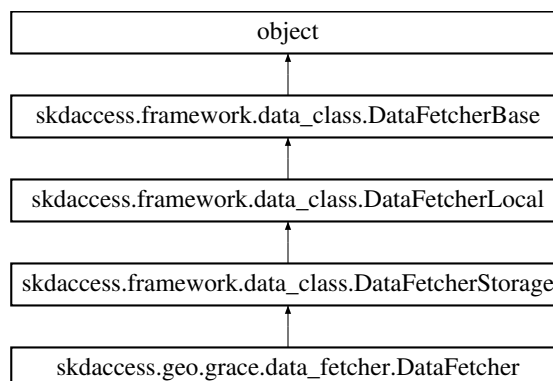
The documentation for this class was generated from the following file:

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

6.35 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 geopooints.
- `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 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`

6.35.1 Detailed Description

Data Fetcher for GRACE data.

6.35.2 Constructor & Destructor Documentation

6.35.2.1 `__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

6.35.3 Member Function Documentation

6.35.3.1 `__str__()`

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

String representation of data fetcher.

Returns

String listing the name and geopoint of data fetcher

6.35.3.2 `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

6.35.3.3 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.35.3.4 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

6.35.3.5 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.35.3.6 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

6.35.3.7 output()

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

Create data wrapper of grace data for specified geopoints.

Returns

Grace Data Wrapper

6.35.3.8 perturb()

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

Perturb parameters.

6.35.3.9 reset()

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

Set all parameters to initial value.

6.35.3.10 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

6.35.3.11 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

6.35.3.12 writeConfig()

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

Write config to disk.

Parameters

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

6.35.4 Member Data Documentation**6.35.4.1 ap_paramList**

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

6.35.4.2 end_date

`skdaccess.geo.grace.DataFetcher.end_date`

6.35.4.3 start_date

`skdaccess.geo.grace.DataFetcher.start_date`

6.35.4.4 verbose

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

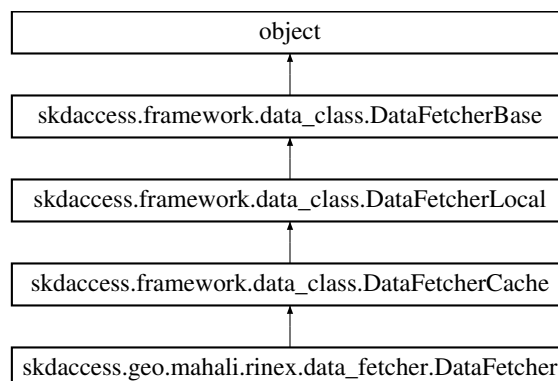
The documentation for this class was generated from the following file:

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

6.36 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 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`

6.36.1 Detailed Description

Data Fetcher for Mahali Data.

6.36.2 Constructor & Destructor Documentation

6.36.2.1 `__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

6.36.3 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()` [1/2]

```
def skdaccess.geo.mahali.rinex.DataFetcher.cacheData (
    self )
```

Downloads all needed data.

Called by [output\(\)](#).

6.36.3.3 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

6.36.3.4 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

6.36.3.5 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

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.mahali.rinex.DataFetcher.output (
    self )
```

Generate data wrapper for Mahali data.

Returns

Mahali 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
-------------	----------------------------------

6.36.4 Member Data Documentation**6.36.4.1 ap_paramList**

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

6.36.4.2 date_range

```
skdaccess.geo.mahali.rinex.DataFetcher.date_range
```

6.36.4.3 end_date

```
skdaccess.geo.mahali.rinex.DataFetcher.end_date
```

6.36.4.4 generate_links

`skdaccess.geo.mahali.rinex.DataFetcher.generate_links`

6.36.4.5 start_date

`skdaccess.geo.mahali.rinex.DataFetcher.start_date`

6.36.4.6 verbose

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

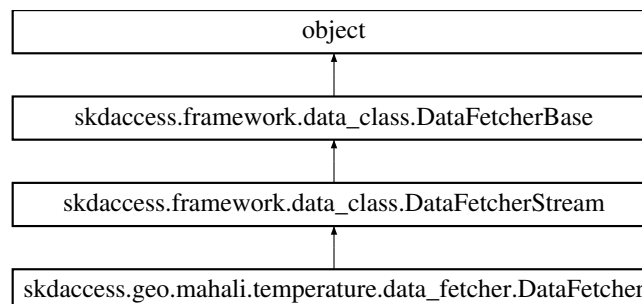
The documentation for this class was generated from the following file:

- [geo/mahali/rinex/data_fetcher.py](#)

6.37 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 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`

6.37.1 Detailed Description

Data Fetcher for Mahali temperature data.

6.37.2 Constructor & Destructor Documentation

6.37.2.1 __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[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)

6.37.3 Member Function Documentation

6.37.3.1 __str__()

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

Generate string description.

6.37.3.2 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.37.3.3 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.37.3.4 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

6.37.3.5 output()

```
def skdaccess.geo.mahali.temperature.DataFetcher.output (
    self )
```

Generate data wrapper for Mahali temperatures.

Returns

Mahali temperature data wrapper

6.37.3.6 perturb()

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

Perturb parameters.

6.37.3.7 reset()

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

Set all parameters to initial value.

6.37.3.8 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

6.37.3.9 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

6.37.3.10 writeConfig()

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

Write config to disk.

Parameters

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

6.37.4 Member Data Documentation

6.37.4.1 ap_paramList

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

6.37.4.2 end_date

`skdaccess.geo.mahali.temperature.DataFetcher.end_date`

6.37.4.3 start_date

`skdaccess.geo.mahali.temperature.DataFetcher.start_date`

6.37.4.4 verbose

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

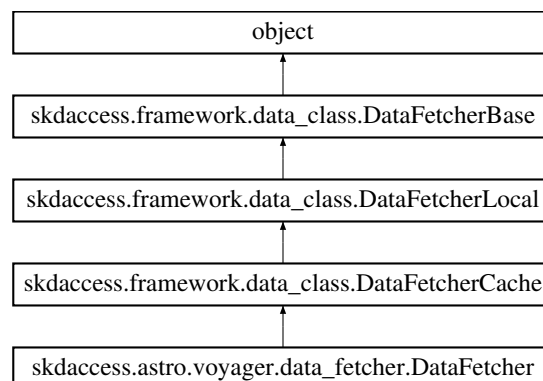
The documentation for this class was generated from the following file:

- [geo/mahali/temperature/data_fetcher.py](#)

6.38 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 `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 `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`

6.38.1 Detailed Description

Data Fetcher for Mahali temperature data.

6.38.2 Constructor & Destructor Documentation

6.38.2.1 `__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).

6.38.3 Member Function Documentation

6.38.3.1 `__str__()`

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

Generate string description.

6.38.3.2 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

6.38.3.3 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

6.38.3.4 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

6.38.3.5 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.38.3.6 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

6.38.3.7 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

6.38.3.8 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.38.3.9 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

6.38.3.10 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

6.38.3.11 output()

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

Generate data wrapper.

Returns

data wrapper of voyager data

6.38.3.12 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

6.38.3.13 parseVoyagerMetadata()

```
def skdaccess.astro.voyager.DataFetcher.parseVoyagerMetadata (
    self,
    in_file )
```

Parse voyager metadata.

Parameters

<i>in_file</i>	Input filename
----------------	----------------

Returns

Dictionary containing metadata

6.38.3.14 perturb()

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

Perturb parameters.

6.38.3.15 reset()

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

Set all parameters to initial value.

6.38.3.16 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

6.38.3.17 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

6.38.3.18 writeConfig()

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

Write config to disk.

Parameters

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

6.38.4 Member Data Documentation

6.38.4.1 ap_paramList

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

6.38.4.2 base_url

skdaccess.astro.voyager.DataFetcher.base_url

6.38.4.3 field_names

skdaccess.astro.voyager.DataFetcher.field_names

6.38.4.4 field_widths

skdaccess.astro.voyager.DataFetcher.field_widths

6.38.4.5 spacecraft_list

skdaccess.astro.voyager.DataFetcher.spacecraft_list

6.38.4.6 verbose

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

6.38.4.7 year_list

`skdaccess.astro.voyager.DataFetcher.year_list`

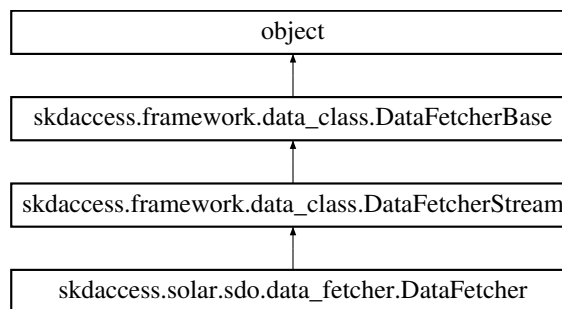
The documentation for this class was generated from the following file:

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

6.39 skdaccess.solar.sdo.DataFetcher Class Reference

Data Fetcher for Mahali temperature data.

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 writeConfig(conf)`
Write config to disk.
- `def verbose_print(self, args, kwargs)`
Print statement if verbose flag is set.

Public Attributes

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

6.39.1 Detailed Description

Data Fetcher for Mahali temperature data.

6.39.2 Constructor & Destructor Documentation

6.39.2.1 `__init__()`

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

Initialize Solar Dynamics Observatory.

Parameters

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

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 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.39.3.4 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

6.39.3.5 output()

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

Generate data wrapper.

Returns

data wrapper of SDO data

6.39.3.6 perturb()

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

Perturb parameters.

6.39.3.7 reset()

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

Set all parameters to initial value.

6.39.3.8 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

6.39.3.9 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

6.39.3.10 writeConfig()

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

Write config to disk.

Parameters

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

6.39.4 Member Data Documentation**6.39.4.1 ap_paramList**

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

6.39.4.2 verbose

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

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

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

6.40.2 Constructor & Destructor Documentation

6.40.2.1 `__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

6.40.3 Member Function Documentation

6.40.3.1 `__str__()`

```
def skdaccess.framework.data_class.DataFetcherBase.__str__ (
    self )
```

Generate string description.

6.40.3.2 `getConfig()`

```
def skdaccess.framework.data_class.DataFetcherBase.getConfig ( )
```

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.40.3.3 getMetadata()

```
def skdaccess.framework.data_class.DataFetcherBase.getMetadata (
    self )
```

Return metadata about Data Fetcher.

Returns

metadata of object.

6.40.3.4 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

6.40.3.5 output()

```
def skdaccess.framework.data_class.DataFetcherBase.output (
    self )
```

Output data wrapper.

Returns

Datawrapper

6.40.3.6 perturb()

```
def skdaccess.framework.data_class.DataFetcherBase.perturb (
    self )
```

Perturb parameters.

6.40.3.7 reset()

```
def skdaccess.framework.data_class.DataFetcherBase.reset (
    self )
```

Set all parameters to initial value.

6.40.3.8 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

6.40.3.9 writeConfig()

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

Write config to disk.

Parameters

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

6.40.4 Member Data Documentation

6.40.4.1 ap_paramList

```
skdaccess.framework.data_class.DataFetcherBase.ap_paramList
```

6.40.4.2 verbose

```
skdaccess.framework.data_class.DataFetcherBase.verbose
```

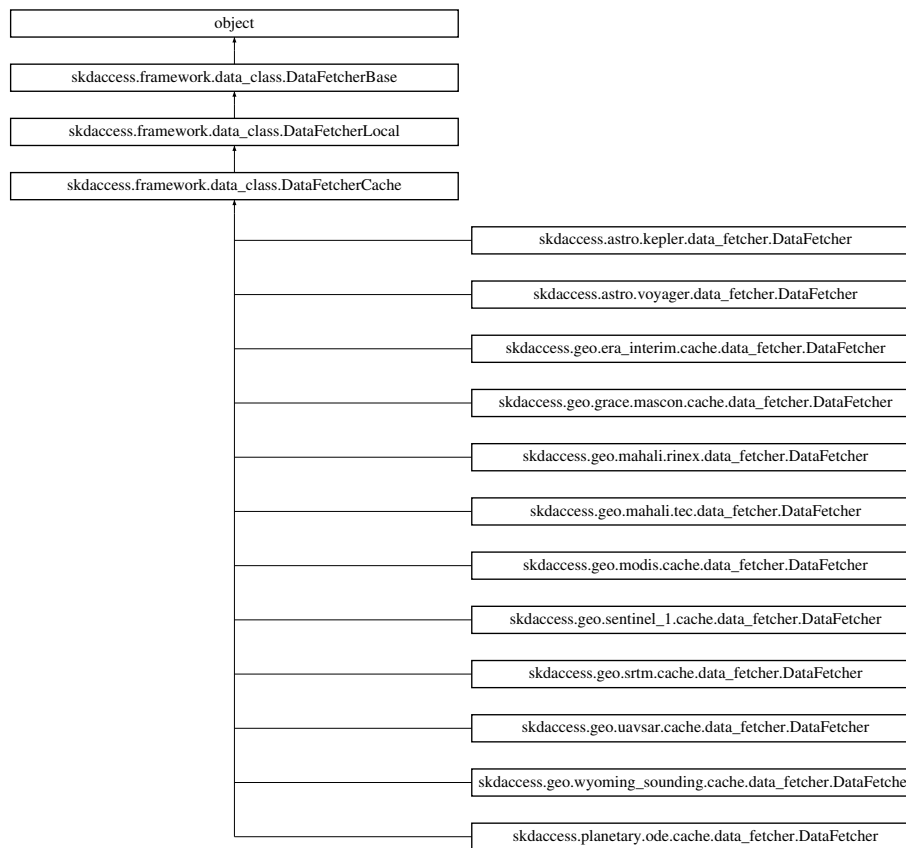
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.41 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 `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 `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 `writeConfig` (conf)
Write config to disk.
- def `verbose_print` (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

- `ap_paramList`
- `verbose`

6.41.1 Detailed Description

Data fetcher base class for downloading data and caching results on hard disk.

6.41.2 Member Function Documentation

6.41.2.1 `__str__()`

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

Generate string description.

6.41.2.2 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

6.41.2.3 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

6.41.2.4 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.41.2.5 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

6.41.2.6 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

6.41.2.7 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.41.2.8 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

6.41.2.9 output()

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

Output data wrapper.

Returns

Datawrapper

6.41.2.10 perturb()

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

Perturb parameters.

6.41.2.11 reset()

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

Set all parameters to initial value.

6.41.2.12 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

6.41.2.13 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

6.41.2.14 writeConfig()

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

Write config to disk.

Parameters

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

6.41.3 Member Data Documentation

6.41.3.1 ap_paramList

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

6.41.3.2 verbose

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

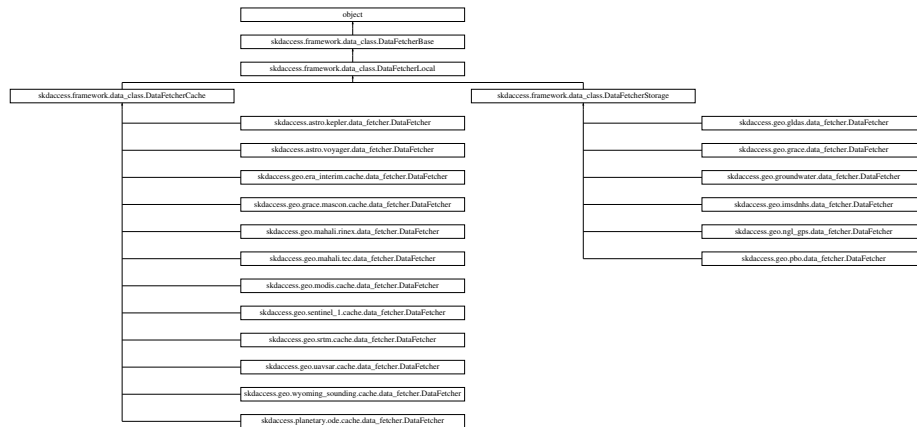
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.42 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 [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](#)

6.42.1 Detailed Description

Data fetcher base class for use when storing data locally.

6.42.2 Member Function Documentation

6.42.2.1 `__str__()`

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

Generate string description.

6.42.2.2 `getConfig()`

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.42.2.3 `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

6.42.2.4 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.42.2.5 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

6.42.2.6 output()

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

Output data wrapper.

Returns

Datawrapper

6.42.2.7 perturb()

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

Perturb parameters.

6.42.2.8 reset()

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

Set all parameters to initial value.

6.42.2.9 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

6.42.2.10 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

6.42.2.11 writeConfig()

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

Write config to disk.

Parameters

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

6.42.3 Member Data Documentation

6.42.3.1 ap_paramList

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

6.42.3.2 verbose

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

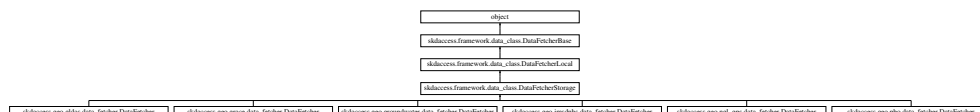
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.43 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 [writeConfig](#) (conf)
Write config to disk.
- def [verbose_print](#) (self, args, kwargs)
Print statement if verbose flag is set.

Public Attributes

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

6.43.1 Detailed Description

Data fetcher base class for use when entire data set is downloaded.

6.43.2 Member Function Documentation

6.43.2.1 `__str__()`

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

Generate string description.

6.43.2.2 `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

6.43.2.3 `getConfig()`

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.43.2.4 `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

6.43.2.5 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.43.2.6 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

6.43.2.7 output()

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

Output data wrapper.

Returns

Datawrapper

6.43.2.8 perturb()

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

Perturb parameters.

6.43.2.9 reset()

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

Set all parameters to initial value.

6.43.2.10 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

6.43.2.11 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

6.43.2.12 writeConfig()

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

Write config to disk.

Parameters

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

6.43.3 Member Data Documentation

6.43.3.1 ap_paramList

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

6.43.3.2 verbose

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

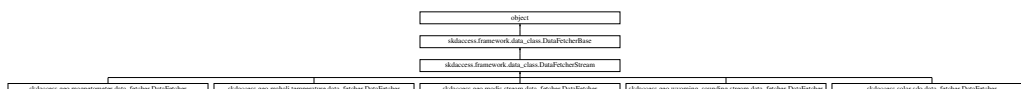
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.44 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 [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 base class for downloading data into memory.

6.44.2 Member Function Documentation

6.44.2.1 [__str__\(\)](#)

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

Generate string description.

6.44.2.2 getConfig()

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

Retrieve skdaccess configuration.

Returns

configParser.ConfigParser object of configuration

6.44.2.3 getMetadata()

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

Return metadata about Data Fetcher.

Returns

metadata of object.

6.44.2.4 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

6.44.2.5 output()

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

Output data wrapper.

Returns

Datawrapper

6.44.2.6 perturb()

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

Perturb parameters.

6.44.2.7 reset()

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

Set all parameters to initial value.

6.44.2.8 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

6.44.2.9 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

6.44.2.10 writeConfig()

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

Write config to disk.

Parameters

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

6.44.3 Member Data Documentation

6.44.3.1 ap_paramList

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

6.44.3.2 verbose

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

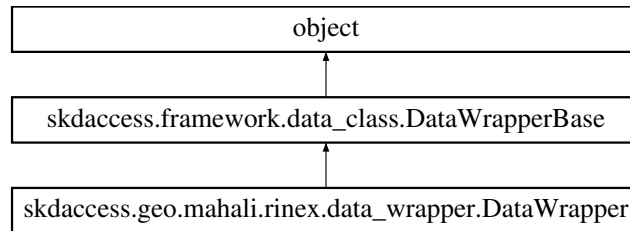
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.45 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](#)

6.45.1 Detailed Description

Data wrapper for Mahali data.

6.45.2 Member Function Documentation

6.45.2.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.45.2.2 `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

6.45.2.3 `get()`

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.45.2.4 getIterator()

```
def skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper.getIterator (
    self )
```

Get iterator to Mahali data.

Returns

Iterator yielding (site,date,nav,obs)

6.45.2.5 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.45.2.6 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.45.2.7 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.45.2.8 reset()

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

Reset data back to original state.

6.45.2.9 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.45.2.10 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.45.3 Member Data Documentation

6.45.3.1 constants

`skdaccess.framework.data_class.DataWrapperBase.constants` [inherited]

6.45.3.2 data

`skdaccess.framework.data_class.DataWrapperBase.data` [inherited]

6.45.3.3 meta_data

`skdaccess.framework.data_class.DataWrapperBase.meta_data` [inherited]

6.45.3.4 results

`skdaccess.framework.data_class.DataWrapperBase.results` [inherited]

6.45.3.5 run_id

`skdaccess.framework.data_class.DataWrapperBase.run_id` [inherited]

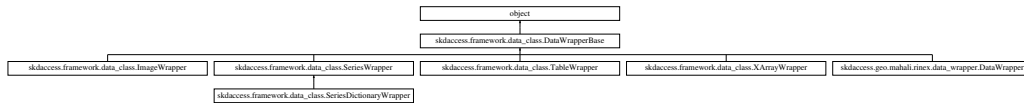
The documentation for this class was generated from the following file:

- [geo/mahali/rinex/data_wrapper.py](#)

6.46 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`

6.46.1 Detailed Description

Base class for wrapping data for use in DiscoveryPipeline.

6.46.2 Constructor & Destructor Documentation

6.46.2.1 `__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

6.46.3 Member Function Documentation

6.46.3.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self )
```

Get length of wrapped data.

Returns

length of wrapped data

6.46.3.2 `addResult()`

```
def skdaccess.framework.data_class.DataWrapperBase.addResult (
    self,
    rkey,
    res )
```

Add a result to the data wrapper.

Parameters

<i>rkey</i>	Result key
<i>rres</i>	Result

6.46.3.3 get()

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self )
```

Retrieve stored data.

Returns

Stored data

6.46.3.4 getIterator()

```
def skdaccess.framework.data_class.DataWrapperBase.getIterator (
    self )
```

Get an iterator to the data.

Returns

iterator to data

6.46.3.5 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self )
```

Retrieve accumulated results, if any.

Returns

store results

6.46.3.6 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self )
```

Get the Run ID.

Returns

run_id

6.46.3.7 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None )
```

Get information about data wrapper.

Returns

The stored metadata

6.46.3.8 reset()

```
def skdaccess.framework.data_class.DataWrapperBase.reset (
    self )
```

Reset data back to original state.

6.46.3.9 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj )
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.46.3.10 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata )
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.46.4 Member Data Documentation

6.46.4.1 constants

```
skdaccess.framework.data_class.DataWrapperBase.constants
```

6.46.4.2 data

```
skdaccess.framework.data_class.DataWrapperBase.data
```

6.46.4.3 meta_data

```
skdaccess.framework.data_class.DataWrapperBase.meta_data
```

6.46.4.4 results

```
skdaccess.framework.data_class.DataWrapperBase.results
```

6.46.4.5 run_id

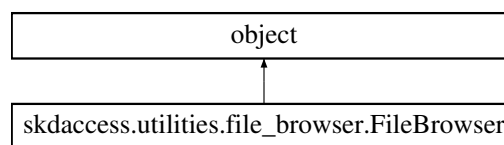
```
skdaccess.framework.data_class.DataWrapperBase.run_id
```

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

- framework/[data_class.py](#)

6.47 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](#)

6.47.1 Constructor & Destructor Documentation

6.47.1.1 __init__()

```
def skdaccess.utilities.file_browser.FileBrowser.__init__ (
    self )
```

6.47.2 Member Function Documentation

6.47.2.1 widget()

```
def skdaccess.utilities.file_browser.FileBrowser.widget (
    self )
```

6.47.3 Member Data Documentation

6.47.3.1 dirs

```
skdaccess.utilities.file_browser.FileBrowser.dirs
```

6.47.3.2 files

```
skdaccess.utilities.file_browser.FileBrowser.files
```

6.47.3.3 path

```
skdaccess.utilities.file_browser.FileBrowser.path
```

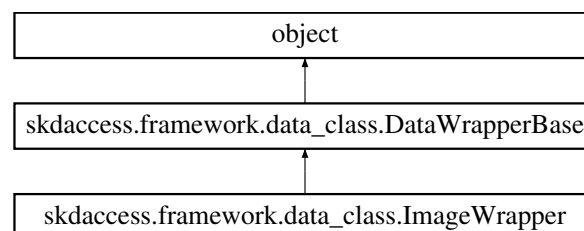
The documentation for this class was generated from the following file:

- [utilities/file_browser.py](#)

6.48 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](#)

6.48.1 Detailed Description

Wrapper for image data.

6.48.2 Member Function Documentation

6.48.2.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.48.2.2 `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

6.48.2.3 `deleteData()`

```
def skdaccess.framework.data_class.ImageWrapper.deleteData (
    self,
    label )
```

Delete image.

Parameters

<i>label</i>	Delete image with label
--------------	-------------------------

6.48.2.4 `get()`

```
def skdaccess.framework.data_class.DataWrapperBase.get (
```

```
self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.48.2.5 getIterator()

```
def skdaccess.framework.data_class.ImageWrapper.getIterator (
    self )
```

Get an iterator to the data.

Returns

Iterator yielding (label, image_data)

6.48.2.6 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.48.2.7 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.48.2.8 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.48.2.9 reset()

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

Reset data back to original state.

6.48.2.10 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.48.2.11 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

6.48.2.12 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.48.3 Member Data Documentation**6.48.3.1 constants**

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.48.3.2 data

```
skdaccess.framework.data_class.DataWrapperBase.data [inherited]
```

6.48.3.3 meta_data

```
skdaccess.framework.data_class.DataWrapperBase.meta_data [inherited]
```

6.48.3.4 results

`skdaccess.framework.data_class.DataWrapperBase.results` [inherited]

6.48.3.5 run_id

`skdaccess.framework.data_class.DataWrapperBase.run_id` [inherited]

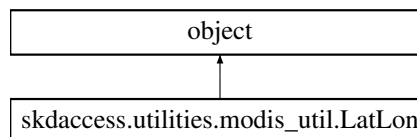
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.49 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](#)

6.49.1 Detailed Description

Calculates Lat/Lon position from y,x pixel coordinate.

6.49.2 Constructor & Destructor Documentation

6.49.2.1 `__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)

6.49.3 Member Function Documentation

6.49.3.1 `__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)

6.49.4 Member Data Documentation**6.49.4.1 alat**

`skdaccess.utilities.modis_util.LatLon.alat`

6.49.4.2 alon

`skdaccess.utilities.modis_util.LatLon.alon`

6.49.4.3 lat_data

`skdaccess.utilities.modis_util.LatLon.lat_data`

6.49.4.4 lon_data

`skdaccess.utilities.modis_util.LatLon.lon_data`

6.49.4.5 x_offset

`skdaccess.utilities.modis_util.LatLon.x_offset`

6.49.4.6 y_offset

`skdaccess.utilities.modis_util.LatLon.y_offset`

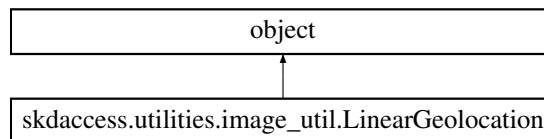
The documentation for this class was generated from the following file:

- [utilities/modis_util.py](#)

6.50 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`

6.50.1 Detailed Description

This class provides functions to convert between pixel and geodetic coordinates.

Assumes a linear relationship between pixel and geodetic coordinates

6.50.2 Constructor & Destructor Documentation

6.50.2.1 __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

6.50.3 Member Function Documentation

6.50.3.1 getExtents()

```
def skdaccess.utilities.image_util.LinearGeolocation.getExtents (
    self )
```

Retrieve the extents of the data.

Returns

(minimum_longitude, maximum_longitude, minimum_latitude, maximum_latitude)

6.50.3.2 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

6.50.3.3 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

6.50.4 Member Data Documentation**6.50.4.1 flip_y**

```
skdaccess.utilities.image_util.LinearGeolocation.flip_y
```

6.50.4.2 lat_extents

```
skdaccess.utilities.image_util.LinearGeolocation.lat_extents
```

6.50.4.3 lat_pixel_size

`skdaccess.utilities.image_util.LinearGeolocation.lat_pixel_size`

6.50.4.4 len_x

`skdaccess.utilities.image_util.LinearGeolocation.len_x`

6.50.4.5 len_y

`skdaccess.utilities.image_util.LinearGeolocation.len_y`

6.50.4.6 lon_extents

`skdaccess.utilities.image_util.LinearGeolocation.lon_extents`

6.50.4.7 lon_pixel_size

`skdaccess.utilities.image_util.LinearGeolocation.lon_pixel_size`

6.50.4.8 start_lat

`skdaccess.utilities.image_util.LinearGeolocation.start_lat`

6.50.4.9 start_lon

`skdaccess.utilities.image_util.LinearGeolocation.start_lon`

6.50.4.10 `x_offset`

`skdaccess.utilities.image_util.LinearGeolocation.x_offset`

6.50.4.11 `y_offset`

`skdaccess.utilities.image_util.LinearGeolocation.y_offset`

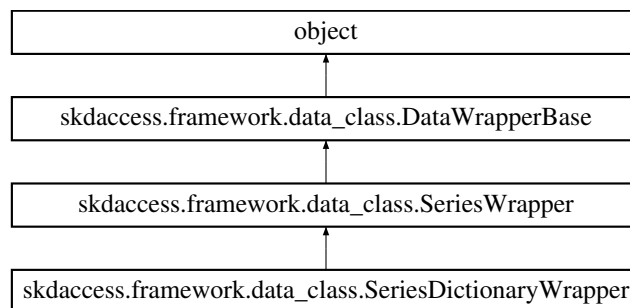
The documentation for this class was generated from the following file:

- [utilities/image_util.py](#)

6.51 `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`

6.51.1 Detailed Description

Data wrapper for series data using a dictionary of data frames.

6.51.2 Member Function Documentation

6.51.2.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.51.2.2 `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

6.51.2.3 get()

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.51.2.4 getIndices()

```
def skdaccess.framework.data_class.SeriesDictionaryWrapper.getIndices (
    self )
```

Get the indices of the data.

Returns

index of data

6.51.2.5 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

6.51.2.6 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

6.51.2.7 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.51.2.8 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.51.2.9 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.51.2.10 reset()

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

Reset data back to original state.

6.51.2.11 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.51.2.12 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.51.3 Member Data Documentation

6.51.3.1 constants

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.51.3.2 data

`skdaccess.framework.data_class.DataWrapperBase.data` [inherited]

6.51.3.3 data_names

`skdaccess.framework.data_class.SeriesWrapper.data_names` [inherited]

6.51.3.4 error_names

`skdaccess.framework.data_class.SeriesWrapper.error_names` [inherited]

6.51.3.5 meta_data

`skdaccess.framework.data_class.DataWrapperBase.meta_data` [inherited]

6.51.3.6 results

`skdaccess.framework.data_class.DataWrapperBase.results` [inherited]

6.51.3.7 run_id

`skdaccess.framework.data_class.DataWrapperBase.run_id` [inherited]

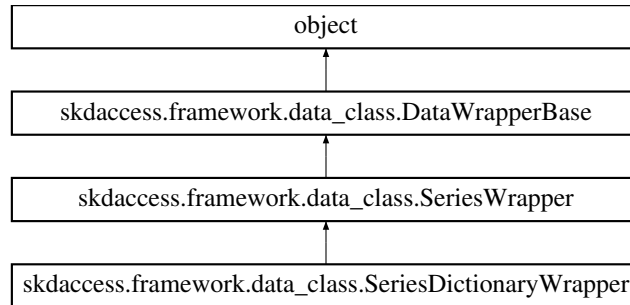
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.52 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](#)

6.52.1 Detailed Description

Data wrapper for series data using a data panel.

6.52.2 Constructor & Destructor Documentation

6.52.2.1 `__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

6.52.3 Member Function Documentation

6.52.3.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.52.3.2 `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

6.52.3.3 `get()`

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.52.3.4 getIndices()

```
def skdaccess.framework.data_class.SeriesWrapper.getIndices (
    self )
```

Get the indices of the data.

Returns

index of data

6.52.3.5 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

6.52.3.6 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

6.52.3.7 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.52.3.8 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.52.3.9 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.52.3.10 reset()

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

Reset data back to original state.

6.52.3.11 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.52.3.12 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.52.4 Member Data Documentation

6.52.4.1 constants

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.52.4.2 data

```
skdaccess.framework.data_class.DataWrapperBase.data [inherited]
```

6.52.4.3 data_names

```
skdaccess.framework.data_class.SeriesWrapper.data_names
```

6.52.4.4 error_names

```
skdaccess.framework.data_class.SeriesWrapper.error_names
```

6.52.4.5 meta_data

```
skdaccess.framework.data_class.DataWrapperBase.meta_data [inherited]
```

6.52.4.6 results

```
skdaccess.framework.data_class.DataWrapperBase.results [inherited]
```

6.52.4.7 run_id

```
skdaccess.framework.data_class.DataWrapperBase.run_id [inherited]
```

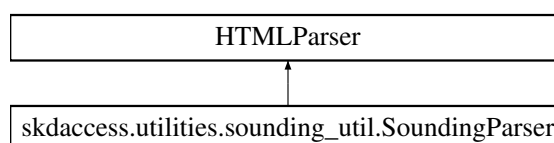
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.53 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](#)

6.53.1 Detailed Description

This class parses Wyoming Sounding data.

6.53.2 Constructor & Destructor Documentation

6.53.2.1 [__init__\(\)](#)

```
def skdaccess.utilities.sounding_util.SoundingParser.__init__ (  
    self )
```

Initialize [SoundingParser](#).

6.53.3 Member Function Documentation

6.53.3.1 [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
-------------	------------

6.53.3.2 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
------------	------------

6.53.3.3 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

6.53.4 Member Data Documentation**6.53.4.1 data_dict**

```
skdaccess.utilities.sounding_util.SoundingParser.data_dict
```

6.53.4.2 in_header

`skdaccess.utilities.sounding_util.SoundingParser.in_header`

6.53.4.3 in_pre_tag

`skdaccess.utilities.sounding_util.SoundingParser.in_pre_tag`

6.53.4.4 label

`skdaccess.utilities.sounding_util.SoundingParser.label`

6.53.4.5 metadata_dict

`skdaccess.utilities.sounding_util.SoundingParser.metadata_dict`

6.53.4.6 read_data

`skdaccess.utilities.sounding_util.SoundingParser.read_data`

6.53.4.7 tmp

`skdaccess.utilities.sounding_util.SoundingParser.tmp`

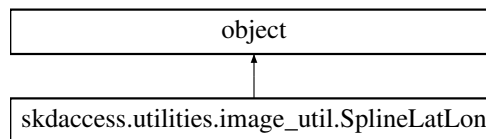
The documentation for this class was generated from the following file:

- [utilities/sounding_util.py](#)

6.54 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](#)

6.54.1 Detailed Description

Holds a 2d spline for interpolating lat/lon grid.

6.54.2 Constructor & Destructor Documentation

6.54.2.1 `__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

6.54.3 Member Function Documentation

6.54.3.1 `__call__()`

```
def skdaccess.utilities.image_util.SplineLatLon.__call__ (
    self,
    y,
    x )
```

Convert pixel coordinates to lat/lon.

Parameters

<i>y</i>	y coordinate
<i>x</i>	x coordinate

Returns

(lat, lon)

6.54.4 Member Data Documentation**6.54.4.1 lat_func**

`skdaccess.utilities.image_util.SplineLatLon.lat_func`

6.54.4.2 lon_func

`skdaccess.utilities.image_util.SplineLatLon.lon_func`

6.54.4.3 x_offset

`skdaccess.utilities.image_util.SplineLatLon.x_offset`

6.54.4.4 y_offset

`skdaccess.utilities.image_util.SplineLatLon.y_offset`

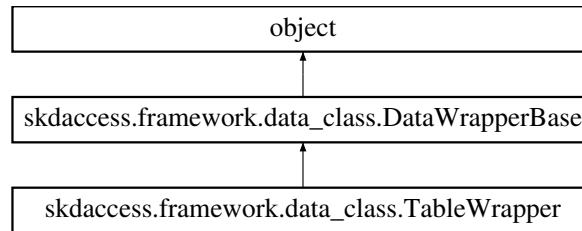
The documentation for this class was generated from the following file:

- [utilities/image_util.py](#)

6.55 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`

6.55.1 Detailed Description

Data wrapper for table data using an ordered dictionary.

6.55.2 Constructor & Destructor Documentation

6.55.2.1 `__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

6.55.3 Member Function Documentation

6.55.3.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.55.3.2 `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

6.55.3.3 `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

6.55.3.4 get()

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.55.3.5 getDefaultColumns()

```
def skdaccess.framework.data_class.TableWrapper.getDefaultColumns (
    self )
```

Get the default columns of data.

Returns

List of default columns

6.55.3.6 getDefaultErrorColumns()

```
def skdaccess.framework.data_class.TableWrapper.getDefaultErrorColumns (
    self )
```

Get the default error columns of data.

Returns

List of default error columns

6.55.3.7 getIterator()

```
def skdaccess.framework.data_class.TableWrapper.getIterator (
    self )
```

Iterator access to data.

Returns

iterator to (label, data frame) from Dictionary

6.55.3.8 getLength()

```
def skdaccess.framework.data_class.TableWrapper.getLength (
    self )
```

Get number of data frames.

Returns

Number of data frames

6.55.3.9 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.55.3.10 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.55.3.11 info()

```
def skdaccess.framework.data_class.DataWrapperBase.info (
    self,
    key = None ) [inherited]
```

Get information about data wrapper.

Returns

The stored metadata

6.55.3.12 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
-------------------	--------------------------

6.55.3.13 reset()

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

Reset data back to original state.

6.55.3.14 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.55.3.15 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

6.55.3.16 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

6.55.3.17 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
```

```
self,  
new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.55.4 Member Data Documentation

6.55.4.1 constants

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.55.4.2 data

```
skdaccess.framework.data_class.DataWrapperBase.data [inherited]
```

6.55.4.3 default_columns

```
skdaccess.framework.data_class.TableWrapper.default_columns
```

6.55.4.4 default_error_columns

```
skdaccess.framework.data_class.TableWrapper.default_error_columns
```

6.55.4.5 meta_data

```
skdaccess.framework.data_class.DataWrapperBase.meta_data [inherited]
```

6.55.4.6 results

```
skdaccess.framework.data_class.DataWrapperBase.results [inherited]
```

6.55.4.7 run_id

```
skdaccess.framework.data_class.DataWrapperBase.run_id [inherited]
```

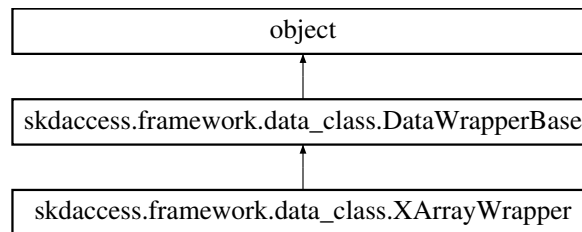
The documentation for this class was generated from the following file:

- framework/[data_class.py](#)

6.56 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](#)

6.56.1 Detailed Description

Wrapper for xarrays.

6.56.2 Constructor & Destructor Documentation

6.56.2.1 `__init__()`

```
def skdaccess.framework.data_class.XArrayWrapper.__init__ (
    self,
    obj_wrap,
    index_list,
    run_id = -1 )
```

6.56.3 Member Function Documentation

6.56.3.1 `__len__()`

```
def skdaccess.framework.data_class.DataWrapperBase.__len__ (
    self ) [inherited]
```

Get length of wrapped data.

Returns

length of wrapped data

6.56.3.2 `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

6.56.3.3 get()

```
def skdaccess.framework.data_class.DataWrapperBase.get (
    self ) [inherited]
```

Retrieve stored data.

Returns

Stored data

6.56.3.4 getIterator()

```
def skdaccess.framework.data_class.XArrayWrapper.getIterator (
    self )
```

Get an iterator that iterates over the index.

Returns

iterator to data

6.56.3.5 getResults()

```
def skdaccess.framework.data_class.DataWrapperBase.getResults (
    self ) [inherited]
```

Retrieve accumulated results, if any.

Returns

store results

6.56.3.6 getRunID()

```
def skdaccess.framework.data_class.DataWrapperBase.getRunID (
    self ) [inherited]
```

Get the Run ID.

Returns

run_id

6.56.3.7 info()

```
def skdaccess.framework.data_class.XArrayWrapper.info (
    self,
    key = None )
```

Get information about xarray data wrapper.

Returns

The stored metadata

6.56.3.8 reset()

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

Reset data back to original state.

6.56.3.9 update()

```
def skdaccess.framework.data_class.DataWrapperBase.update (
    self,
    obj ) [inherited]
```

Updated wrapped data.

Parameters

<i>obj</i>	New data for wrapper
------------	----------------------

6.56.3.10 updateMetadata()

```
def skdaccess.framework.data_class.DataWrapperBase.updateMetadata (
    self,
    new_metadata ) [inherited]
```

Update metadata.

Parameters

<i>new_metadata</i>	New metadata
---------------------	--------------

6.56.4 Member Data Documentation

6.56.4.1 constants

```
skdaccess.framework.data_class.DataWrapperBase.constants [inherited]
```

6.56.4.2 data

```
skdaccess.framework.data_class.DataWrapperBase.data [inherited]
```

6.56.4.3 index_list

```
skdaccess.framework.data_class.XArrayWrapper.index_list
```

6.56.4.4 meta_data

`skdaccess.framework.data_class.DataWrapperBase.meta_data` [inherited]

6.56.4.5 results

`skdaccess.framework.data_class.DataWrapperBase.results` [inherited]

6.56.4.6 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 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](#)

7.2 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](#)

7.3 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](#)

7.4 solar/sdo/data_fetcher.py File Reference

Classes

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

Namespaces

- [skdaccess.solar.sdo.data_fetcher](#)

7.5 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](#)

7.6 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](#)

7.7 geo/grace/data_fetcher.py File Reference

Classes

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

Namespaces

- [skdaccess.geo.grace.data_fetcher](#)

7.8 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](#)

7.9 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](#)

7.10 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](#)

7.11 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](#)

7.12 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/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](#)

7.14 geo/gldas/data_fetcher.py File Reference

Classes

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

Namespaces

- [skdaccess.geo.gldas.data_fetcher](#)

7.15 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](#)

7.16 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](#)

7.17 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](#)

7.18 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](#)

7.19 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](#)

7.20 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](#)

7.21 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](#)

7.22 `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](#)

7.23 `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](#)

7.24 `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](#)

7.25 `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](#)

7.26 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](#)

7.27 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](#)

7.28 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/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](#)

7.30 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](#)

7.31 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](#)

7.32 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](#)

7.33 utilities/file_browser.py File Reference

Classes

- class [skdaccess.utilities.file_browser.FileBrowser](#)

Namespaces

- [skdaccess.utilities.file_browser](#)

7.34 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)

7.35 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.

7.36 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](#)

7.37 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 by the median for the quarter.

7.38 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.parselonoFile](#) (in_file, compression='infer')

7.39 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.

7.40 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.

7.41 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.

7.42 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.

7.43 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.

7.44 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.

7.45 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)

7.46 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`, [58](#)
`skdaccess::framework::param_class::AutoListCycle`,
[63](#)
`skdaccess::framework::param_class::AutoList`↔
 `Permute`, [67](#)
`skdaccess::framework::param_class::AutoList`↔
 `Remove`, [71](#)
`skdaccess::framework::param_class::AutoList`↔
 `Subset`, [75](#)
`skdaccess::framework::param_class::AutoParam`, [80](#)
`skdaccess::framework::param_class::AutoParamList`,
[82](#)
`skdaccess::framework::param_class::AutoParam`↔
 `ListCycle`, [85](#)
`skdaccess::framework::param_class::AutoParam`↔
 `MinMax`, [88](#)
`skdaccess::utilities::image_util::SplineLatLon`, [333](#)
`skdaccess::utilities::modis_util::LatLon`, [310](#)

`__getitem__`

`skdaccess::framework::param_class::AutoList`, [58](#)
`skdaccess::framework::param_class::AutoListCycle`,
[63](#)
`skdaccess::framework::param_class::AutoList`↔
 `Permute`, [67](#)
`skdaccess::framework::param_class::AutoList`↔
 `Remove`, [71](#)
`skdaccess::framework::param_class::AutoList`↔
 `Subset`, [75](#)

`__init__`

`skdaccess::astro::kepler::data_fetcher::DataFetcher`,
[217](#)
`skdaccess::astro::voyager::data_fetcher::Data`↔
 `Fetcher`, [254](#)
`skdaccess::framework::data_class::DataFetcher`↔
 `Base`, [268](#)
`skdaccess::framework::data_class::DataWrapper`↔
 `Base`, [298](#)
`skdaccess::framework::data_class::SeriesWrapper`,
[323](#)
`skdaccess::framework::data_class::TableWrapper`,
[336](#)
`skdaccess::framework::data_class::XArrayWrapper`,
[344](#)
`skdaccess::framework::param_class::AutoList`, [58](#)

`skdaccess::framework::param_class::AutoListCycle`,
[62](#)
`skdaccess::framework::param_class::AutoList`↔
 `Remove`, [71](#)
`skdaccess::framework::param_class::AutoParam`, [79](#)
`skdaccess::framework::param_class::AutoParamList`,
[82](#)
`skdaccess::framework::param_class::AutoParam`↔
 `ListCycle`, [85](#)
`skdaccess::framework::param_class::AutoParam`↔
 `MinMax`, [87](#)
`skdaccess::geo::era_interim::cache::data_fetcher::`↔
 `DataFetcher`, [196](#)
`skdaccess::geo::gldas::data_fetcher::DataFetcher`,
[91](#)
`skdaccess::geo::grace::data_fetcher::DataFetcher`,
[234](#)
`skdaccess::geo::grace::mascon::cache::data_`↔
 `fetcher::DataFetcher`, [182](#)
`skdaccess::geo::groundwater::data_fetcher::Data`↔
 `Fetcher`, [106](#)
`skdaccess::geo::imsdnhs::data_fetcher::Data`↔
 `Fetcher`, [189](#)
`skdaccess::geo::magnetometer::data_fetcher::`↔
 `DataFetcher`, [128](#)
`skdaccess::geo::mahali::rinex::data_fetcher::Data`↔
 `Fetcher`, [241](#)
`skdaccess::geo::mahali::tec::data_fetcher::Data`↔
 `Fetcher`, [210](#)
`skdaccess::geo::mahali::temperature::data_fetcher`↔
 `::DataFetcher`, [248](#)
`skdaccess::geo::modis::cache::cloud_mask::data_`↔
 `fetcher::DataFetcher`, [150](#)
`skdaccess::geo::modis::cache::cloud_opacity`↔
 `::data_fetcher::DataFetcher`, [149](#)
`skdaccess::geo::modis::cache::data_fetcher::Data`↔
 `Fetcher`, [154](#)
`skdaccess::geo::modis::cache::reflectance::data_`↔
 `fetcher::DataFetcher`, [152](#)
`skdaccess::geo::modis::stream::cloud_mask::data`↔
 `_fetcher::DataFetcher`, [164](#)
`skdaccess::geo::modis::stream::cloud_opacity`↔
 `::data_fetcher::DataFetcher`, [163](#)
`skdaccess::geo::modis::stream::data_fetcher::`↔
 `DataFetcher`, [175](#)

- skdaccess::geo::modis::stream::reflectance::data_↵
fetcher::DataFetcher, 134
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
203
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 226
- skdaccess::geo::sentinel_1::cache::data_fetcher::↵
DataFetcher, 98
- skdaccess::geo::srtm::cache::data_fetcher::Data↵
Fetcher, 113
- skdaccess::geo::uavsar::cache::data_fetcher::Data↵
Fetcher, 121
- skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 136
- skdaccess::geo::wyoming_sounding::stream::data_↵
fetcher::DataFetcher, 144
- skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 166
- skdaccess::solar::sdo::data_fetcher::DataFetcher,
263
- skdaccess::utilities::file_browser::FileBrowser, 302
- skdaccess::utilities::image_util::AffineGlobalCoords,
55
- skdaccess::utilities::image_util::LinearGeolocation,
313
- skdaccess::utilities::image_util::SplineLatLon, 332
- skdaccess::utilities::modis_util::LatLon, 310
- skdaccess::utilities::sounding_util::SoundingParser,
329
- __len__
 - skdaccess::framework::data_class::DataWrapper↵
Base, 298
 - skdaccess::framework::data_class::ImageWrapper,
304
 - skdaccess::framework::data_class::SeriesDictionary↵
Wrapper, 317
 - skdaccess::framework::data_class::SeriesWrapper,
323
 - skdaccess::framework::data_class::TableWrapper,
337
 - skdaccess::framework::data_class::XArrayWrapper,
344
 - skdaccess::framework::param_class::AutoList, 59
 - skdaccess::framework::param_class::AutoListCycle,
63
 - skdaccess::framework::param_class::AutoList↵
Permute, 67
 - skdaccess::framework::param_class::AutoList↵
Remove, 72
 - skdaccess::framework::param_class::AutoList↵
Subset, 76
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↵
Wrapper, 293
- __setitem__
 - skdaccess::framework::param_class::AutoList, 59
 - skdaccess::framework::param_class::AutoListCycle,
64
 - skdaccess::framework::param_class::AutoList↵
Permute, 68
 - skdaccess::framework::param_class::AutoList↵
Remove, 72
 - skdaccess::framework::param_class::AutoList↵
Subset, 76
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
218
 - skdaccess::astro::voyager::data_fetcher::Data↵
Fetcher, 254
 - skdaccess::framework::data_class::DataFetcher↵
Base, 268
 - skdaccess::framework::data_class::DataFetcher↵
Cache, 272
 - skdaccess::framework::data_class::DataFetcher↵
Local, 279
 - skdaccess::framework::data_class::DataFetcher↵
Storage, 283
 - skdaccess::framework::data_class::DataFetcher↵
Stream, 288
 - skdaccess::framework::param_class::AutoList, 59
 - skdaccess::framework::param_class::AutoListCycle,
64
 - skdaccess::framework::param_class::AutoList↵
Permute, 68
 - skdaccess::framework::param_class::AutoList↵
Remove, 72
 - skdaccess::framework::param_class::AutoList↵
Subset, 76
 - skdaccess::framework::param_class::AutoParam, 80
 - skdaccess::framework::param_class::AutoParamList,
82
 - skdaccess::framework::param_class::AutoParam↵
ListCycle, 85
 - skdaccess::framework::param_class::AutoParam↵
MinMax, 88
 - skdaccess::geo::era_interim::cache::data_fetcher::↵
DataFetcher, 196
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
92
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
235
 - skdaccess::geo::grace::mascon::cache::data_↵
fetcher::DataFetcher, 182
 - skdaccess::geo::groundwater::data_fetcher::Data↵
Fetcher, 106
 - skdaccess::geo::imdsdnh::data_fetcher::Data↵
Fetcher, 190
 - skdaccess::geo::magnetometer::data_fetcher::↵
DataFetcher, 129
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↵

- Fetcher, [241](#)
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
 - Fetcher, [210](#)
 - skdaccess::geo::mahali::temperature::data_fetcher↔
 - ::DataFetcher, [249](#)
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
 - Fetcher, [155](#)
 - skdaccess::geo::modis::stream::data_fetcher::↔
 - DataFetcher, [176](#)
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
[203](#)
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, [227](#)
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
 - DataFetcher, [98](#)
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
 - Fetcher, [113](#)
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
 - Fetcher, [122](#)
 - skdaccess::geo::wyoming_sounding::cache::data_↔
 - fetcher::DataFetcher, [137](#)
 - skdaccess::geo::wyoming_sounding::stream::data_↔
 - _fetcher::DataFetcher, [145](#)
 - skdaccess::planetary::ode::cache::data_fetcher::↔
 - DataFetcher, [167](#)
 - skdaccess::solar::sdo::data_fetcher::DataFetcher,
[263](#)
- addColumn
 - skdaccess::framework::data_class::TableWrapper,
[337](#)
- addResult
 - skdaccess::framework::data_class::DataWrapper↔
 - Base, [298](#)
 - skdaccess::framework::data_class::ImageWrapper,
[305](#)
 - skdaccess::framework::data_class::SeriesDictionary↔
 - Wrapper, [317](#)
 - skdaccess::framework::data_class::SeriesWrapper,
[324](#)
 - skdaccess::framework::data_class::TableWrapper,
[337](#)
 - skdaccess::framework::data_class::XArrayWrapper,
[344](#)
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
 - Wrapper, [293](#)
- alat
 - skdaccess::utilities::modis_util::LatLon, [311](#)
- alon
 - skdaccess::utilities::modis_util::LatLon, [311](#)
- antenna_info
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, [232](#)
- ap_paramList
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
[224](#)
- skdaccess::astro::voyager::data_fetcher::Data↔
 - Fetcher, [261](#)
- skdaccess::framework::data_class::DataFetcher↔
 - Base, [270](#)
- skdaccess::framework::data_class::DataFetcher↔
 - Cache, [277](#)
- skdaccess::framework::data_class::DataFetcher↔
 - Local, [282](#)
- skdaccess::framework::data_class::DataFetcher↔
 - Storage, [287](#)
- skdaccess::framework::data_class::DataFetcher↔
 - Stream, [291](#)
- skdaccess::geo::era_interim::cache::data_fetcher::↔
 - DataFetcher, [200](#)
- skdaccess::geo::gldas::data_fetcher::DataFetcher,
[95](#)
- skdaccess::geo::grace::data_fetcher::DataFetcher,
[238](#)
- skdaccess::geo::grace::mascon::cache::data_↔
 - fetcher::DataFetcher, [187](#)
- skdaccess::geo::groundwater::data_fetcher::Data↔
 - Fetcher, [110](#)
- skdaccess::geo::imsdnhs::data_fetcher::Data↔
 - Fetcher, [193](#)
- skdaccess::geo::magnetometer::data_fetcher::↔
 - DataFetcher, [132](#)
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔
 - Fetcher, [246](#)
- skdaccess::geo::mahali::tec::data_fetcher::Data↔
 - Fetcher, [215](#)
- skdaccess::geo::mahali::temperature::data_fetcher↔
 - ::DataFetcher, [251](#)
- skdaccess::geo::modis::cache::data_fetcher::Data↔
 - Fetcher, [160](#)
- skdaccess::geo::modis::stream::data_fetcher::↔
 - DataFetcher, [178](#)
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
[207](#)
- skdaccess::geo::pbo::data_fetcher::DataFetcher, [232](#)
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
 - DataFetcher, [103](#)
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
 - Fetcher, [118](#)
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
 - Fetcher, [126](#)
- skdaccess::geo::wyoming_sounding::cache::data_↔
 - fetcher::DataFetcher, [141](#)
- skdaccess::geo::wyoming_sounding::stream::data_↔
 - _fetcher::DataFetcher, [147](#)
- skdaccess::planetary::ode::cache::data_fetcher::↔
 - DataFetcher, [171](#)
- skdaccess::solar::sdo::data_fetcher::DataFetcher,
[266](#)
- arcsecond_sampling

- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 118
- astro/kepler/data_fetcher.py, 358
- astro/voyager/data_fetcher.py, 358
- averageDates
 - skdaccess::utilities::grace_util, 27
- base_url
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 261
- cacheData
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
218
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 254
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 272
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 196
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 182
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 241
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 211
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 155
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 98
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 114
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 122
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 137
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 167
- calibrateModis
 - skdaccess::utilities::modis_util, 36
- channels
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 132
- checkBit
 - skdaccess::utilities::modis_util, 37
- checkIfDataExists
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
220
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 255
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 273
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 197
- skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 183
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 242
- skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 211
- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 156
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 99
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 114
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 123
- skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 137
- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 167
- combine_water_heights
 - skdaccess::utilities::gw_util, 31
- computeEWD
 - skdaccess::utilities::grace_util, 28
- constants
 - skdaccess::framework::data_class::DataWrapper↔
Base, 301
 - skdaccess::framework::data_class::ImageWrapper,
308
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 320
 - skdaccess::framework::data_class::SeriesWrapper,
327
 - skdaccess::framework::data_class::TableWrapper,
342
 - skdaccess::framework::data_class::XArrayWrapper,
347
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 296
- convert_date
 - skdaccess::utilities::mahali_util, 35
- convertBinCentersToEdges
 - skdaccess::utilities::image_util, 32
- convertToStr
 - skdaccess::utilities::support, 52
- coordinate_dict
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 193
- correct_CRISM_label
 - skdaccess::utilities::ode_util, 41
- correct_file_name_case_in_label
 - skdaccess::utilities::ode_util, 42
- correct_label_file
 - skdaccess::utilities::ode_util, 42
- createGrid
 - skdaccess::utilities::modis_util, 37

- current_index
 - skdaccess::framework::param_class::AutoParam↔
ListCycle, 86
- cutoff
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 110
- data
 - skdaccess::framework::data_class::DataWrapper↔
Base, 301
 - skdaccess::framework::data_class::ImageWrapper,
308
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 320
 - skdaccess::framework::data_class::SeriesWrapper,
327
 - skdaccess::framework::data_class::TableWrapper,
342
 - skdaccess::framework::data_class::XArrayWrapper,
347
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 296
- data_dict
 - skdaccess::utilities::sounding_util::SoundingParser,
330
- data_names
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 321
 - skdaccess::framework::data_class::SeriesWrapper,
327
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 200
- data_type
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 132
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
207
- date_list
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 201
- date_range
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 246
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 215
- dateMismatch
 - skdaccess::utilities::grace_util, 28
- day_end
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 141
 - skdaccess::geo::wyoming_sounding::stream::data_↔
_fetcher::DataFetcher, 148
- day_start
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 141
 - skdaccess::geo::wyoming_sounding::stream::data_↔
_fetcher::DataFetcher, 148
- daynightboth
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 160
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 179
- decimals
 - skdaccess::framework::param_class::AutoParam↔
MinMax, 89
- default_columns
 - skdaccess::framework::data_class::TableWrapper,
342
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 232
- default_error_columns
 - skdaccess::framework::data_class::TableWrapper,
342
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 232
- deleteData
 - skdaccess::framework::data_class::ImageWrapper,
305
- dirs
 - skdaccess::utilities::file_browser::FileBrowser, 303
- downloadFullDataset
 - skdaccess::framework::data_class::DataFetcher↔
Storage, 284
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
92
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
235
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 107
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 190
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
203
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 227
- downloadKeplerData
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
220
- eastern_lon
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 171
- end_date
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
95
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
238
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 187

- skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 110
- skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 194
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 246
- skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 215
- skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 252
- skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 161
- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 179
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
207
- end_hour
 - skdaccess::geo::wyoming_sounding::cache::data↔
fetcher::DataFetcher, 142
 - skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, 148
- end_time
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 132
- error_names
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 321
 - skdaccess::framework::data_class::SeriesWrapper,
327
- field_names
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 261
- field_widths
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 261
- file_name
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 171
- files
 - skdaccess::utilities::file_browser::FileBrowser, 303
- find_data
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 157
- flip_y
 - skdaccess::utilities::image_util::LinearGeolocation,
314
- framework/data_class.py, 349
- framework/param_class.py, 350
- generate_links
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 246
- generateQueries
 - skdaccess::utilities::sounding_util, 50
- generateURL
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 256
- geo/era_interim/cache/data_fetcher.py, 353
- geo/gldas/data_fetcher.py, 353
- geo/grace/data_fetcher.py, 351
- geo/grace/mascon/cache/data_fetcher.py, 351
- geo/groundwater/data_fetcher.py, 358
- geo/imsdnhs/data_fetcher.py, 353
- geo/magnetometer/data_fetcher.py, 354
- geo/mahali/rinex/data_fetcher.py, 352
- geo/mahali/rinex/data_wrapper.py, 350
- geo/mahali/tec/data_fetcher.py, 352
- geo/mahali/temperature/data_fetcher.py, 352
- geo/modis/cache/cloud_mask/data_fetcher.py, 355
- geo/modis/cache/cloud_opacity/data_fetcher.py, 355
- geo/modis/cache/data_fetcher.py, 356
- geo/modis/cache/reflectance/data_fetcher.py, 355
- geo/modis/stream/cloud_mask/data_fetcher.py, 356
- geo/modis/stream/cloud_opacity/data_fetcher.py, 356
- geo/modis/stream/data_fetcher.py, 357
- geo/modis/stream/reflectance/data_fetcher.py, 356
- geo/ngl_gps/data_fetcher.py, 352
- geo/pbo/data_fetcher.py, 358
- geo/sentinel_1/cache/data_fetcher.py, 354
- geo/srtm/cache/data_fetcher.py, 357
- geo/uavsar/cache/data_fetcher.py, 357
- geo/wyoming_sounding/cache/data_fetcher.py, 354
- geo/wyoming_sounding/stream/data_fetcher.py, 354
- get
 - skdaccess::framework::data_class::DataWrapper↔
Base, 299
 - skdaccess::framework::data_class::ImageWrapper,
305
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 318
 - skdaccess::framework::data_class::SeriesWrapper,
324
 - skdaccess::framework::data_class::TableWrapper,
338
 - skdaccess::framework::data_class::XArrayWrapper,
345
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 293
- get_files_urls
 - skdaccess::utilities::ode_util, 42
- get_query_url
 - skdaccess::utilities::ode_util, 42
- get_raster_array
 - skdaccess::utilities::ode_util, 43
- get_raster_extent
 - skdaccess::utilities::ode_util, 43
- getAllOptions
 - skdaccess::framework::param_class::AutoList, 60

- skdaccess::framework::param_class::AutoListCycle, 64
- skdaccess::framework::param_class::AutoList↔
Permute, 68
- skdaccess::framework::param_class::AutoList↔
Remove, 73
- skdaccess::framework::param_class::AutoList↔
Subset, 77
- getAntennaLogs
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 204
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 228
- getConfig
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 221
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 256
 - skdaccess::framework::data_class::DataFetcher↔
Base, 268
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 274
 - skdaccess::framework::data_class::DataFetcher↔
Local, 279
 - skdaccess::framework::data_class::DataFetcher↔
Storage, 284
 - skdaccess::framework::data_class::DataFetcher↔
Stream, 288
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 197
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 93
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 236
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 184
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 107
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 191
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 129
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 243
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 212
 - skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 249
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 157
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 176
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 204
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 228
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 100
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 115
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 123
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 138
 - skdaccess::geo::wyoming_sounding::stream::data_↔
fetcher::DataFetcher, 145
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 168
 - skdaccess::solar::sdo::data_fetcher::DataFetcher, 263
- getDataLocation
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 221
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 256
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 274
 - skdaccess::framework::data_class::DataFetcher↔
Local, 279
 - skdaccess::framework::data_class::DataFetcher↔
Storage, 284
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 197
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 93
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 236
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 184
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 107
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 191
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 243
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 212
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 157
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 204
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 228
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 100
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 115
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 123
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 138

- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 168
- getDataMetadata
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 129
- getDefaultColumns
 - skdaccess::framework::data_class::TableWrapper,
338
- getDefaultErrorColumns
 - skdaccess::framework::data_class::TableWrapper,
338
- getExtents
 - skdaccess::utilities::image_util::LinearGeolocation,
313
- getExtentsFromCentersPlateCarree
 - skdaccess::utilities::image_util, 32
- getFileIDs
 - skdaccess::utilities::modis_util, 38
- getFileURLs
 - skdaccess::utilities::modis_util, 39
- getGeoTransform
 - skdaccess::utilities::image_util, 33
- getHDFStorage
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
221
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 257
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 274
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 198
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 184
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 243
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 212
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 158
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 100
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 115
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 124
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 138
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 168
- getImageType
 - skdaccess::utilities::modis_util, 39
- getIndices
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 318
- skdaccess::framework::data_class::SeriesWrapper,
324
- getInfo
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 228
- getIterator
 - skdaccess::framework::data_class::DataWrapper↔
Base, 299
 - skdaccess::framework::data_class::ImageWrapper,
306
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 318
 - skdaccess::framework::data_class::SeriesWrapper,
325
 - skdaccess::framework::data_class::TableWrapper,
338
 - skdaccess::framework::data_class::XArrayWrapper,
345
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 294
- getLatLon
 - skdaccess::utilities::image_util::LinearGeolocation,
313
- getLatLonRange
 - skdaccess::utilities::pbo_util, 45
- getLength
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 318
 - skdaccess::framework::data_class::SeriesWrapper,
325
 - skdaccess::framework::data_class::TableWrapper,
339
- getMasconPlacement
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 185
- getMetadata
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
222
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 257
 - skdaccess::framework::data_class::DataFetcher↔
Base, 268
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 275
 - skdaccess::framework::data_class::DataFetcher↔
Local, 280
 - skdaccess::framework::data_class::DataFetcher↔
Storage, 285
 - skdaccess::framework::data_class::DataFetcher↔
Stream, 289
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 198
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
93
 - skdaccess::geo::grace::data_fetcher::DataFetcher,

- 236
- skdaccess::geo::grace::mascon::cache::data_↵
fetcher::DataFetcher, 185
- skdaccess::geo::groundwater::data_fetcher::Data_↵
Fetcher, 108
- skdaccess::geo::imsdnhs::data_fetcher::Data_↵
Fetcher, 191
- skdaccess::geo::magnetometer::data_fetcher::↵
DataFetcher, 130
- skdaccess::geo::mahali::rinex::data_fetcher::Data_↵
Fetcher, 244
- skdaccess::geo::mahali::tec::data_fetcher::Data_↵
Fetcher, 213
- skdaccess::geo::mahali::temperature::data_fetcher_↵
::DataFetcher, 249
- skdaccess::geo::modis::cache::data_fetcher::Data_↵
Fetcher, 158
- skdaccess::geo::modis::stream::data_fetcher::↵
DataFetcher, 176
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
205
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 229
- skdaccess::geo::sentinel_1::cache::data_fetcher::↵
DataFetcher, 101
- skdaccess::geo::srtm::cache::data_fetcher::Data_↵
Fetcher, 116
- skdaccess::geo::uavsar::cache::data_fetcher::Data_↵
Fetcher, 124
- skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 139
- skdaccess::geo::wyoming_sounding::stream::data_↵
_fetcher::DataFetcher, 145
- skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 169
- skdaccess::solar::sdo::data_fetcher::DataFetcher,
264
- getMetadataFiles
 - skdaccess::astro::voyager::data_fetcher::Data_↵
Fetcher, 257
- getModisData
 - skdaccess::utilities::modis_util, 40
- getPixelYX
 - skdaccess::utilities::image_util::AffineGlobalCoords,
56
- getProjectedYX
 - skdaccess::utilities::image_util::AffineGlobalCoords,
56
- getROlstations
 - skdaccess::utilities::pbo_util, 46
- getResults
 - skdaccess::framework::data_class::DataWrapper_↵
Base, 299
 - skdaccess::framework::data_class::ImageWrapper,
306
 - skdaccess::framework::data_class::SeriesDictionary_↵
Wrapper, 319
 - skdaccess::framework::data_class::SeriesWrapper,
325
 - skdaccess::framework::data_class::TableWrapper,
339
 - skdaccess::framework::data_class::XArrayWrapper,
345
 - skdaccess::geo::mahali::rinex::data_wrapper::Data_↵
Wrapper, 294
- getRunID
 - skdaccess::framework::data_class::DataWrapper_↵
Base, 299
 - skdaccess::framework::data_class::ImageWrapper,
306
 - skdaccess::framework::data_class::SeriesDictionary_↵
Wrapper, 319
 - skdaccess::framework::data_class::SeriesWrapper,
325
 - skdaccess::framework::data_class::TableWrapper,
339
 - skdaccess::framework::data_class::XArrayWrapper,
345
 - skdaccess::geo::mahali::rinex::data_wrapper::Data_↵
Wrapper, 294
- getSRTMData
 - skdaccess::utilities::srtm_util, 51
- getSRTMLatLon
 - skdaccess::utilities::srtm_util, 51
- getStartEndDate
 - skdaccess::utilities::grace_util, 30
- getStationCoords
 - skdaccess::utilities::pbo_util, 46
- getStationMetadata
 - skdaccess::geo::groundwater::data_fetcher::Data_↵
Fetcher, 108
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
205
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 229
- getYX
 - skdaccess::utilities::image_util::LinearGeolocation,
314
- grid
 - skdaccess::geo::modis::cache::data_fetcher::Data_↵
Fetcher, 161
 - skdaccess::geo::modis::stream::data_fetcher::↵
DataFetcher, 179
- grid_fill
 - skdaccess::geo::modis::cache::data_fetcher::Data_↵
Fetcher, 161
 - skdaccess::geo::modis::stream::data_fetcher::↵
DataFetcher, 179
- handle_data

- skdaccess::utilities::sounding_util::SoundingParser, 329
- handle_endtag
 - skdaccess::utilities::sounding_util::SoundingParser, 330
- handle_starttag
 - skdaccess::utilities::sounding_util::SoundingParser, 330
- in_header
 - skdaccess::utilities::sounding_util::SoundingParser, 330
- in_pre_tag
 - skdaccess::utilities::sounding_util::SoundingParser, 331
- index
 - skdaccess::framework::param_class::AutoListCycle, 65
- index_date_only
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 232
- index_list
 - skdaccess::framework::data_class::XArrayWrapper, 347
- info
 - skdaccess::framework::data_class::DataWrapper↔Base, 300
 - skdaccess::framework::data_class::ImageWrapper, 306
 - skdaccess::framework::data_class::SeriesDictionary↔Wrapper, 319
 - skdaccess::framework::data_class::SeriesWrapper, 326
 - skdaccess::framework::data_class::TableWrapper, 339
 - skdaccess::framework::data_class::XArrayWrapper, 346
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔Wrapper, 294
- instrument
 - skdaccess::planetary::ode::cache::data_fetcher::↔DataFetcher, 172
- interval
 - skdaccess::geo::magnetometer::data_fetcher::↔DataFetcher, 133
- label
 - skdaccess::utilities::sounding_util::SoundingParser, 331
- lat_data
 - skdaccess::utilities::modis_util::LatLon, 311
- lat_extents
 - skdaccess::utilities::image_util::LinearGeolocation, 314
- lat_func
 - skdaccess::utilities::image_util::SplineLatLon, 334
- lat_pixel_size
 - skdaccess::utilities::image_util::LinearGeolocation, 314
- lat_range
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 208
- lat_spline
 - skdaccess::utilities::image_util, 34
- lat_tile_end
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔Fetcher, 118
- lat_tile_start
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔Fetcher, 119
- len_x
 - skdaccess::utilities::image_util::LinearGeolocation, 315
- len_y
 - skdaccess::utilities::image_util::LinearGeolocation, 315
- list_val_list
 - skdaccess::framework::param_class::AutoListCycle, 65
- llh_url
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔Fetcher, 126
- local_paths
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔DataFetcher, 103
- lon_data
 - skdaccess::utilities::modis_util::LatLon, 311
- lon_extents
 - skdaccess::utilities::image_util::LinearGeolocation, 315
- lon_func
 - skdaccess::utilities::image_util::SplineLatLon, 334
- lon_pixel_size
 - skdaccess::utilities::image_util::LinearGeolocation, 315
- lon_range
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 208
- lon_spline
 - skdaccess::utilities::image_util, 34
- lon_tile_end
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔Fetcher, 119
- lon_tile_start
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔Fetcher, 119
- mascon_placement_url
 - skdaccess::geo::grace::mascon::cache::data↔fetcher::DataFetcher, 187

- mascon_url
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 188
- mask_water
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 119
- max_lat
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 172
- max_ob_time
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 172
- mdyratio
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
208
- memmap
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 127
- merge_srtm_tiles
 - skdaccess::utilities::srtm_util, 52
- meta_data
 - skdaccess::framework::data_class::DataWrapper↔
Base, 301
 - skdaccess::framework::data_class::ImageWrapper,
308
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 321
 - skdaccess::framework::data_class::SeriesWrapper,
328
 - skdaccess::framework::data_class::TableWrapper,
342
 - skdaccess::framework::data_class::XArrayWrapper,
347
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 296
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 232
- metadata_dict
 - skdaccess::utilities::sounding_util::SoundingParser,
331
- metadata_url_list
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 127
- min_lat
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 172
- min_ob_time
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 172
- mission
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 172
- modis_id
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 161
- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 179
- modis_identifier
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 161
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 179
- modis_platform
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 161
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 179
- month_list
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 142
 - skdaccess::geo::wyoming_sounding::stream::data_↔
fetcher::DataFetcher, 148
- multirun_enabled
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
222
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 258
 - skdaccess::framework::data_class::DataFetcher↔
Base, 269
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 275
 - skdaccess::framework::data_class::DataFetcher↔
Local, 280
 - skdaccess::framework::data_class::DataFetcher↔
Storage, 285
 - skdaccess::framework::data_class::DataFetcher↔
Stream, 289
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 198
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
94
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
236
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 185
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 108
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 191
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 130
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 244
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 213
 - skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 249
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 158

- skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 176
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
205
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 229
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 101
- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 116
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 124
- skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 139
- skdaccess::geo::wyoming_sounding::stream::data_↔
_fetcher::DataFetcher, 145
- skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 169
- skdaccess::solar::sdo::data_fetcher::DataFetcher,
264
- n
 - skdaccess::framework::param_class::AutoList↔
Remove, 74
 - skdaccess::framework::param_class::AutoParam↔
MinMax, 89
- n_max
 - skdaccess::framework::param_class::AutoParam↔
MinMax, 89
- normalize
 - skdaccess::utilities::kepler_util, 35
- nostab_sys
 - skdaccess::utilities::pbo_util, 47
- number_product_limit
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 172
- output
 - skdaccess::astro::kepler::data_fetcher::DataFetcher,
222
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 258
 - skdaccess::framework::data_class::DataFetcher↔
Base, 269
 - skdaccess::framework::data_class::DataFetcher↔
Cache, 275
 - skdaccess::framework::data_class::DataFetcher↔
Local, 280
 - skdaccess::framework::data_class::DataFetcher↔
Storage, 285
 - skdaccess::framework::data_class::DataFetcher↔
Stream, 289
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 199
 - skdaccess::geo::gldas::data_fetcher::DataFetcher,
94
 - skdaccess::geo::grace::data_fetcher::DataFetcher,
237
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, 185
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, 108
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, 192
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, 130
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, 244
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, 213
 - skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, 250
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 159
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 177
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher,
205
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 229
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 101
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 116
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, 125
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, 139
 - skdaccess::geo::wyoming_sounding::stream::data_↔
_fetcher::DataFetcher, 145, 146
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 169
 - skdaccess::solar::sdo::data_fetcher::DataFetcher,
264
- parselonoFile
 - skdaccess::utilities::mahali_util, 36
- parseSatelliteData
 - skdaccess::utilities::sentinel_1_util, 49
- parseVoyagerData
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 258
- parseVoyagerMetadata
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, 259
- password
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 201
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 103

- skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [119](#)
- path
 - skdaccess::utilities::file_browser::FileBrowser, [303](#)
- perturb
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, [222](#)
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [259](#)
 - skdaccess::framework::data_class::DataFetcher↔
Base, [269](#)
 - skdaccess::framework::data_class::DataFetcher↔
Cache, [275](#)
 - skdaccess::framework::data_class::DataFetcher↔
Local, [280](#)
 - skdaccess::framework::data_class::DataFetcher↔
Storage, [285](#)
 - skdaccess::framework::data_class::DataFetcher↔
Stream, [289](#)
 - skdaccess::framework::param_class::AutoList, [60](#)
 - skdaccess::framework::param_class::AutoListCycle, [64](#)
 - skdaccess::framework::param_class::AutoList↔
Permute, [69](#)
 - skdaccess::framework::param_class::AutoList↔
Remove, [73](#)
 - skdaccess::framework::param_class::AutoList↔
Subset, [77](#)
 - skdaccess::framework::param_class::AutoParam, [80](#)
 - skdaccess::framework::param_class::AutoParamList, [83](#)
 - skdaccess::framework::param_class::AutoParam↔
ListCycle, [85](#)
 - skdaccess::framework::param_class::AutoParam↔
MinMax, [88](#)
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, [199](#)
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, [94](#)
 - skdaccess::geo::grace::data_fetcher::DataFetcher, [237](#)
 - skdaccess::geo::grace::mascon::cache::data_↔
fetcher::DataFetcher, [186](#)
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, [109](#)
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, [192](#)
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, [130](#)
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, [244](#)
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, [213](#)
 - skdaccess::geo::mahali::temperature::data_fetcher↔
↔DataFetcher, [250](#)
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [159](#)
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, [177](#)
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, [206](#)
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, [230](#)
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [101](#)
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [116](#)
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, [125](#)
 - skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, [140](#)
 - skdaccess::geo::wyoming_sounding::stream::data_↔
_fetcher::DataFetcher, [146](#)
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [170](#)
 - skdaccess::solar::sdo::data_fetcher::DataFetcher, [264](#)
 - planetary/ode/cache/data_fetcher.py, [351](#)
 - polarization
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, [103](#)
 - product_id
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [173](#)
 - product_type
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, [173](#)
 - progress_bar
 - skdaccess::utilities::support, [53](#)
 - propagateErrors
 - skdaccess::utilities::pbo_util, [47](#)
 - quarter_list
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, [224](#)
 - query_files_urls
 - skdaccess::utilities::ode_util, [44](#)
 - query_yes_no
 - skdaccess::utilities::ode_util, [45](#)
 - read_data
 - skdaccess::utilities::sounding_util::SoundingParser, [331](#)
 - readMODISData
 - skdaccess::utilities::modis_util, [40](#)
 - readTellusData
 - skdaccess::utilities::grace_util, [30](#)
 - readUAVSARMetadata
 - skdaccess::utilities::uavsar_util, [54](#)
 - remove_ndv

- skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 173
- removeAntennaOffset
 - skdaccess::utilities::pbo_util, 48
- removeFrames
 - skdaccess::framework::data_class::TableWrapper, 340
- resample
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 96
- rescale
 - skdaccess::utilities::modis_util, 40
- reset
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 223
 - skdaccess::astro::voyager::data_fetcher::DataFetcher, 259
 - skdaccess::framework::data_class::DataFetcherBase, 269
 - skdaccess::framework::data_class::DataFetcherCache, 276
 - skdaccess::framework::data_class::DataFetcherLocal, 281
 - skdaccess::framework::data_class::DataFetcherStorage, 286
 - skdaccess::framework::data_class::DataFetcherStream, 290
 - skdaccess::framework::data_class::DataWrapperBase, 300
 - skdaccess::framework::data_class::ImageWrapper, 307
 - skdaccess::framework::data_class::SeriesDictionaryWrapper, 319
 - skdaccess::framework::data_class::SeriesWrapper, 326
 - skdaccess::framework::data_class::TableWrapper, 340
 - skdaccess::framework::data_class::XArrayWrapper, 346
 - skdaccess::framework::param_class::AutoList, 60
 - skdaccess::framework::param_class::AutoListCycle, 65
 - skdaccess::framework::param_class::AutoListPermute, 69
 - skdaccess::framework::param_class::AutoListRemove, 73
 - skdaccess::framework::param_class::AutoListSubset, 77
 - skdaccess::framework::param_class::AutoParam, 80
 - skdaccess::framework::param_class::AutoParamList, 83
 - skdaccess::framework::param_class::AutoParamListCycle, 86
 - skdaccess::framework::param_class::AutoParamMinMax, 89
- skdaccess::geo::era_interim::cache::data_fetcher::DataFetcher, 199
- skdaccess::geo::gldas::data_fetcher::DataFetcher, 94
- skdaccess::geo::grace::data_fetcher::DataFetcher, 237
- skdaccess::geo::grace::mascon::cache::data_fetcher::DataFetcher, 186
- skdaccess::geo::groundwater::data_fetcher::DataFetcher, 109
- skdaccess::geo::imsdnhs::data_fetcher::DataFetcher, 192
- skdaccess::geo::magnetometer::data_fetcher::DataFetcher, 131
- skdaccess::geo::mahali::rinex::data_fetcher::DataFetcher, 245
- skdaccess::geo::mahali::rinex::data_wrapper::DataWrapper, 295
- skdaccess::geo::mahali::tec::data_fetcher::DataFetcher, 214
- skdaccess::geo::mahali::temperature::data_fetcher::DataFetcher, 250
- skdaccess::geo::modis::cache::data_fetcher::DataFetcher, 159
- skdaccess::geo::modis::stream::data_fetcher::DataFetcher, 177
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 206
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 230
- skdaccess::geo::sentinel_1::cache::data_fetcher::DataFetcher, 102
- skdaccess::geo::srtm::cache::data_fetcher::DataFetcher, 117
- skdaccess::geo::uavsar::cache::data_fetcher::DataFetcher, 125
- skdaccess::geo::wyoming_sounding::cache::data_fetcher::DataFetcher, 140
- skdaccess::geo::wyoming_sounding::stream::data_fetcher::DataFetcher, 146
- skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 170
- skdaccess::solar::sdo::data_fetcher::DataFetcher, 265
- result_offset_number
 - skdaccess::planetary::ode::cache::data_fetcher::DataFetcher, 173
- results
 - skdaccess::framework::data_class::DataWrapperBase, 301
 - skdaccess::framework::data_class::ImageWrapper, 308
 - skdaccess::framework::data_class::SeriesDictionaryWrapper, 321

- skdaccess::framework::data_class::SeriesWrapper, 328
- skdaccess::framework::data_class::TableWrapper, 342
- skdaccess::framework::data_class::XArrayWrapper, 348
- skdaccess::geo::mahali::rinex::data_wrapper::Data↔Wrapper, 296
- retrieveCommonDatesHDF
 - skdaccess::utilities::support, 53
- retrieveOnlineData
 - skdaccess::framework::data_class::DataFetcher↔Stream, 290
 - skdaccess::geo::magnetometer::data_fetcher::↔DataFetcher, 131
 - skdaccess::geo::mahali::temperature::data_fetcher↔::DataFetcher, 250
 - skdaccess::geo::modis::stream::data_fetcher::↔DataFetcher, 177
 - skdaccess::geo::wyoming_sounding::stream::data↔_fetcher::DataFetcher, 146
 - skdaccess::solar::sdo::data_fetcher::DataFetcher, 265
- run_id
 - skdaccess::framework::data_class::DataWrapper↔Base, 302
 - skdaccess::framework::data_class::ImageWrapper, 309
 - skdaccess::framework::data_class::SeriesDictionary↔Wrapper, 321
 - skdaccess::framework::data_class::SeriesWrapper, 328
 - skdaccess::framework::data_class::TableWrapper, 343
 - skdaccess::framework::data_class::XArrayWrapper, 348
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔Wrapper, 296
- satellite_url_list
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔DataFetcher, 104
- scale_factor_url
 - skdaccess::geo::grace::mascon::cache::data↔_fetcher::DataFetcher, 188
- setDataLocation
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 223
 - skdaccess::astro::voyager::data_fetcher::Data↔Fetcher, 259
 - skdaccess::framework::data_class::DataFetcher↔Cache, 276
 - skdaccess::framework::data_class::DataFetcher↔Local, 281
- skdaccess::framework::data_class::DataFetcher↔Storage, 286
- skdaccess::geo::era_interim::cache::data_fetcher::↔DataFetcher, 199
- skdaccess::geo::gldas::data_fetcher::DataFetcher, 94
- skdaccess::geo::grace::data_fetcher::DataFetcher, 237
- skdaccess::geo::grace::mascon::cache::data↔_fetcher::DataFetcher, 186
- skdaccess::geo::groundwater::data_fetcher::Data↔Fetcher, 109
- skdaccess::geo::imsdnhs::data_fetcher::Data↔Fetcher, 192
- skdaccess::geo::mahali::rinex::data_fetcher::Data↔Fetcher, 245
- skdaccess::geo::mahali::tec::data_fetcher::Data↔Fetcher, 214
- skdaccess::geo::modis::cache::data_fetcher::Data↔Fetcher, 159
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 206
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 230
- skdaccess::geo::sentinel_1::cache::data_fetcher::↔DataFetcher, 102
- skdaccess::geo::srtm::cache::data_fetcher::Data↔Fetcher, 117
- skdaccess::geo::uavsar::cache::data_fetcher::Data↔Fetcher, 125
- skdaccess::geo::wyoming_sounding::cache::data↔_fetcher::DataFetcher, 140
- skdaccess::planetary::ode::cache::data_fetcher::↔DataFetcher, 170
- setStationList
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 231
- skdaccess, 13
- skdaccess.astro, 13
- skdaccess.astro.kepler, 13
- skdaccess.astro.kepler.data_fetcher, 13
- skdaccess.astro.kepler.DataFetcher, 216
- skdaccess.astro.voyager, 14
- skdaccess.astro.voyager.data_fetcher, 14
- skdaccess.astro.voyager.DataFetcher, 252
- skdaccess.framework, 14
- skdaccess.framework.data_class, 14
- skdaccess.framework.data_class.DataFetcherBase, 267
- skdaccess.framework.data_class.DataFetcherCache, 271
- skdaccess.framework.data_class.DataFetcherLocal, 278
- skdaccess.framework.data_class.DataFetcherStorage, 282
- skdaccess.framework.data_class.DataFetcherStream, 287
- skdaccess.framework.data_class.DataWrapperBase, 297
- skdaccess.framework.data_class.ImageWrapper, 303

- skdaccess.framework.data_class.SeriesDictionary↔
Wrapper, 316
- skdaccess.framework.data_class.SeriesWrapper, 322
- skdaccess.framework.data_class.TableWrapper, 335
- skdaccess.framework.data_class.XArrayWrapper, 343
- skdaccess.framework.param_class, 15
- skdaccess.framework.param_class.AutoList, 57
- skdaccess.framework.param_class.AutoListCycle, 61
- skdaccess.framework.param_class.AutoListPermute, 66
- skdaccess.framework.param_class.AutoListRemove, 70
- skdaccess.framework.param_class.AutoListSubset, 74
- skdaccess.framework.param_class.AutoParam, 78
- skdaccess.framework.param_class.AutoParamList, 81
- skdaccess.framework.param_class.AutoParamListCycle, 84
- skdaccess.framework.param_class.AutoParamMinMax, 87
- skdaccess.geo, 15
- skdaccess.geo.era_interim, 15
- skdaccess.geo.era_interim.cache, 16
- skdaccess.geo.era_interim.cache.data_fetcher, 16
- skdaccess.geo.era_interim.cache.DataFetcher, 194
- skdaccess.geo.gldas, 16
- skdaccess.geo.gldas.data_fetcher, 16
- skdaccess.geo.gldas.DataFetcher, 90
- skdaccess.geo.grace, 16
- skdaccess.geo.grace.data_fetcher, 16
- skdaccess.geo.grace.DataFetcher, 233
- skdaccess.geo.grace.mascon, 17
- skdaccess.geo.grace.mascon.cache, 17
- skdaccess.geo.grace.mascon.cache.data_fetcher, 17
- skdaccess.geo.grace.mascon.cache.DataFetcher, 180
- skdaccess.geo.groundwater, 17
- skdaccess.geo.groundwater.data_fetcher, 17
- skdaccess.geo.groundwater.DataFetcher, 105
- skdaccess.geo.imsdnhs, 17
- skdaccess.geo.imsdnhs.data_fetcher, 18
- skdaccess.geo.imsdnhs.DataFetcher, 188
- skdaccess.geo.magnetometer, 18
- skdaccess.geo.magnetometer.data_fetcher, 18
- skdaccess.geo.magnetometer.DataFetcher, 127
- skdaccess.geo.mahali, 18
- skdaccess.geo.mahali.rinex, 18
- skdaccess.geo.mahali.rinex.data_fetcher, 19
- skdaccess.geo.mahali.rinex.data_wrapper, 19
- skdaccess.geo.mahali.rinex.data_wrapper.DataWrapper, 292
- skdaccess.geo.mahali.rinex.DataFetcher, 239
- skdaccess.geo.mahali.tec, 19
- skdaccess.geo.mahali.tec.data_fetcher, 19
- skdaccess.geo.mahali.tec.DataFetcher, 209
- skdaccess.geo.mahali.temperature, 19
- skdaccess.geo.mahali.temperature.data_fetcher, 19
- skdaccess.geo.mahali.temperature.DataFetcher, 247
- skdaccess.geo.modis, 20
- skdaccess.geo.modis.cache, 20
- skdaccess.geo.modis.cache.cloud_mask, 20
- skdaccess.geo.modis.cache.cloud_mask.data_fetcher, 20
- skdaccess.geo.modis.cache.cloud_mask.DataFetcher, 150
- skdaccess.geo.modis.cache.cloud_opacity, 20
- skdaccess.geo.modis.cache.cloud_opacity.data_fetcher, 20
- skdaccess.geo.modis.cache.cloud_opacity.DataFetcher, 149
- skdaccess.geo.modis.cache.data_fetcher, 21
- skdaccess.geo.modis.cache.DataFetcher, 152
- skdaccess.geo.modis.cache.reflectance, 21
- skdaccess.geo.modis.cache.reflectance.data_fetcher, 21
- skdaccess.geo.modis.cache.reflectance.DataFetcher, 151
- skdaccess.geo.modis.stream, 21
- skdaccess.geo.modis.stream.cloud_mask, 21
- skdaccess.geo.modis.stream.cloud_mask.data_fetcher, 21
- skdaccess.geo.modis.stream.cloud_mask.DataFetcher, 163
- skdaccess.geo.modis.stream.cloud_opacity, 22
- skdaccess.geo.modis.stream.cloud_opacity.data_fetcher, 22
- skdaccess.geo.modis.stream.cloud_opacity.DataFetcher, 162
- skdaccess.geo.modis.stream.data_fetcher, 22
- skdaccess.geo.modis.stream.DataFetcher, 174
- skdaccess.geo.modis.stream.reflectance, 22
- skdaccess.geo.modis.stream.reflectance.data_fetcher, 22
- skdaccess.geo.modis.stream.reflectance.DataFetcher, 133
- skdaccess.geo.ngl_gps, 22
- skdaccess.geo.ngl_gps.data_fetcher, 23
- skdaccess.geo.ngl_gps.DataFetcher, 201
- skdaccess.geo.pbo, 23
- skdaccess.geo.pbo.data_fetcher, 23
- skdaccess.geo.pbo.DataFetcher, 225
- skdaccess.geo.sentinel_1, 23
- skdaccess.geo.sentinel_1.cache, 23
- skdaccess.geo.sentinel_1.cache.data_fetcher, 23
- skdaccess.geo.sentinel_1.cache.DataFetcher, 96
- skdaccess.geo.srtm, 24
- skdaccess.geo.srtm.cache, 24
- skdaccess.geo.srtm.cache.data_fetcher, 24
- skdaccess.geo.srtm.cache.DataFetcher, 111
- skdaccess.geo.uavsar, 24
- skdaccess.geo.uavsar.cache, 24
- skdaccess.geo.uavsar.cache.data_fetcher, 24
- skdaccess.geo.uavsar.cache.DataFetcher, 120
- skdaccess.geo.wyoming_sounding, 25
- skdaccess.geo.wyoming_sounding.cache, 25

- skdaccess.geo.wyoming_sounding.cache.data_fetcher, 25
- skdaccess.geo.wyoming_sounding.cache.DataFetcher, 134
- skdaccess.geo.wyoming_sounding.stream, 25
- skdaccess.geo.wyoming_sounding.stream.data_fetcher, 25
- skdaccess.geo.wyoming_sounding.stream.DataFetcher, 143
- skdaccess.planetary, 25
- skdaccess.planetary.ode, 26
- skdaccess.planetary.ode.cache, 26
- skdaccess.planetary.ode.cache.data_fetcher, 26
- skdaccess.planetary.ode.cache.DataFetcher, 164
- skdaccess.solar, 26
- skdaccess.solar.sdo, 26
- skdaccess.solar.sdo.data_fetcher, 26
- skdaccess.solar.sdo.DataFetcher, 262
- skdaccess.utilities, 27
- skdaccess.utilities.file_browser, 27
- skdaccess.utilities.file_browser.FileBrowser, 302
- skdaccess.utilities.grace_util, 27
- skdaccess.utilities.gw_util, 31
- skdaccess.utilities.image_util, 31
- skdaccess.utilities.image_util.AffineGlobalCoords, 55
- skdaccess.utilities.image_util.LinearGeolocation, 312
- skdaccess.utilities.image_util.SplineLatLon, 332
- skdaccess.utilities.kepler_util, 35
- skdaccess.utilities.mahali_util, 35
- skdaccess.utilities.modis_util, 36
- skdaccess.utilities.modis_util.LatLon, 309
- skdaccess.utilities.ode_util, 41
- skdaccess.utilities.pbo_util, 45
- skdaccess.utilities.sentinel_1_util, 49
- skdaccess.utilities.sounding_util, 50
- skdaccess.utilities.sounding_util.SoundingParser, 328
- skdaccess.utilities.srtm_util, 51
- skdaccess.utilities.support, 52
- skdaccess.utilities.uavsar_util, 53
- skdaccess::astro::kepler::data_fetcher::DataFetcher
 - __init__, 217
 - __str__, 218
 - ap_paramList, 224
 - cacheData, 218
 - checkIfDataExists, 220
 - downloadKeplerData, 220
 - getConfig, 221
 - getDataLocation, 221
 - getHDFStorage, 221
 - getMetadata, 222
 - multirun_enabled, 222
 - output, 222
 - perturb, 222
 - quarter_list, 224
 - reset, 223
 - setDataLocation, 223
 - verbose, 224
 - verbose_print, 223
 - writeConfig, 224
- skdaccess::astro::voyager::data_fetcher::DataFetcher
 - __init__, 254
 - __str__, 254
 - ap_paramList, 261
 - base_url, 261
 - cacheData, 254
 - checkIfDataExists, 255
 - field_names, 261
 - field_widths, 261
 - generateURL, 256
 - getConfig, 256
 - getDataLocation, 256
 - getHDFStorage, 257
 - getMetadata, 257
 - getMetadataFiles, 257
 - multirun_enabled, 258
 - output, 258
 - parseVoyagerData, 258
 - parseVoyagerMetadata, 259
 - perturb, 259
 - reset, 259
 - setDataLocation, 259
 - spacecraft_list, 261
 - verbose, 261
 - verbose_print, 260
 - writeConfig, 260
 - year_list, 261
- skdaccess::framework::data_class::DataFetcherBase
 - __init__, 268
 - __str__, 268
 - ap_paramList, 270
 - getConfig, 268
 - getMetadata, 268
 - multirun_enabled, 269
 - output, 269
 - perturb, 269
 - reset, 269
 - verbose, 270
 - verbose_print, 270
 - writeConfig, 270
- skdaccess::framework::data_class::DataFetcherCache
 - __str__, 272
 - ap_paramList, 277
 - cacheData, 272
 - checkIfDataExists, 273
 - getConfig, 274
 - getDataLocation, 274
 - getHDFStorage, 274
 - getMetadata, 275

- multirun_enabled, 275
 - output, 275
 - perturb, 275
 - reset, 276
 - setDataLocation, 276
 - verbose, 277
 - verbose_print, 276
 - writeConfig, 277
- skdaccess::framework::data_class::DataFetcherLocal
 - __str__, 279
 - ap_paramList, 282
 - getConfig, 279
 - getDataLocation, 279
 - getMetadata, 280
 - multirun_enabled, 280
 - output, 280
 - perturb, 280
 - reset, 281
 - setDataLocation, 281
 - verbose, 282
 - verbose_print, 281
 - writeConfig, 282
- skdaccess::framework::data_class::DataFetcherStorage
 - __str__, 283
 - ap_paramList, 287
 - downloadFullDataset, 284
 - getConfig, 284
 - getDataLocation, 284
 - getMetadata, 285
 - multirun_enabled, 285
 - output, 285
 - perturb, 285
 - reset, 286
 - setDataLocation, 286
 - verbose, 287
 - verbose_print, 286
 - writeConfig, 287
- skdaccess::framework::data_class::DataFetcherStream
 - __str__, 288
 - ap_paramList, 291
 - getConfig, 288
 - getMetadata, 289
 - multirun_enabled, 289
 - output, 289
 - perturb, 289
 - reset, 290
 - retrieveOnlineData, 290
 - verbose, 291
 - verbose_print, 290
 - writeConfig, 291
- skdaccess::framework::data_class::DataWrapperBase
 - __init__, 298
 - __len__, 298
 - addResult, 298
 - constants, 301
 - data, 301
 - get, 299
 - getIterator, 299
 - getResults, 299
 - getRunID, 299
 - info, 300
 - meta_data, 301
 - reset, 300
 - results, 301
 - run_id, 302
 - update, 300
 - updateMetadata, 301
- skdaccess::framework::data_class::ImageWrapper
 - __len__, 304
 - addResult, 305
 - constants, 308
 - data, 308
 - deleteData, 305
 - get, 305
 - getIterator, 306
 - getResults, 306
 - getRunID, 306
 - info, 306
 - meta_data, 308
 - reset, 307
 - results, 308
 - run_id, 309
 - update, 307
 - updateData, 307
 - updateMetadata, 308
- skdaccess::framework::data_class::SeriesDictionary↔
 - Wrapper
 - __len__, 317
 - addResult, 317
 - constants, 320
 - data, 320
 - data_names, 321
 - error_names, 321
 - get, 318
 - getIndices, 318
 - getIterator, 318
 - getLength, 318
 - getResults, 319
 - getRunID, 319
 - info, 319
 - meta_data, 321
 - reset, 319
 - results, 321
 - run_id, 321
 - update, 320
 - updateMetadata, 320
- skdaccess::framework::data_class::SeriesWrapper
 - __init__, 323

- `__len__`, 323
- `addResult`, 324
- `constants`, 327
- `data`, 327
- `data_names`, 327
- `error_names`, 327
- `get`, 324
- `getIndices`, 324
- `getIterator`, 325
- `getLength`, 325
- `getResults`, 325
- `getRunID`, 325
- `info`, 326
- `meta_data`, 328
- `reset`, 326
- `results`, 328
- `run_id`, 328
- `update`, 326
- `updateMetadata`, 327
- `skdaccess::framework::data_class::TableWrapper`
 - `__init__`, 336
 - `__len__`, 337
 - `addColumn`, 337
 - `addResult`, 337
 - `constants`, 342
 - `data`, 342
 - `default_columns`, 342
 - `default_error_columns`, 342
 - `get`, 338
 - `getDefaultColumns`, 338
 - `getDefaultErrorColumns`, 338
 - `getIterator`, 338
 - `getLength`, 339
 - `getResults`, 339
 - `getRunID`, 339
 - `info`, 339
 - `meta_data`, 342
 - `removeFrames`, 340
 - `reset`, 340
 - `results`, 342
 - `run_id`, 343
 - `update`, 340
 - `updateData`, 341
 - `updateFrames`, 341
 - `updateMetadata`, 341
- `skdaccess::framework::data_class::XArrayWrapper`
 - `__init__`, 344
 - `__len__`, 344
 - `addResult`, 344
 - `constants`, 347
 - `data`, 347
 - `get`, 345
 - `getIterator`, 345
 - `getResults`, 345
 - `getRunID`, 345
 - `index_list`, 347
 - `info`, 346
 - `meta_data`, 347
 - `reset`, 346
 - `results`, 348
 - `run_id`, 348
 - `update`, 346
 - `updateMetadata`, 347
- `skdaccess::framework::param_class::AutoList`
 - `__call__`, 58
 - `__getitem__`, 58
 - `__init__`, 58
 - `__len__`, 59
 - `__setitem__`, 59
 - `__str__`, 59
 - `getAllOptions`, 60
 - `perturb`, 60
 - `reset`, 60
 - `val`, 60
 - `val_init`, 61
 - `val_list`, 61
- `skdaccess::framework::param_class::AutoListCycle`
 - `__call__`, 63
 - `__getitem__`, 63
 - `__init__`, 62
 - `__len__`, 63
 - `__setitem__`, 64
 - `__str__`, 64
 - `getAllOptions`, 64
 - `index`, 65
 - `list_val_list`, 65
 - `perturb`, 64
 - `reset`, 65
 - `val`, 65
 - `val_init`, 65
 - `val_list`, 66
- `skdaccess::framework::param_class::AutoListPermute`
 - `__call__`, 67
 - `__getitem__`, 67
 - `__len__`, 67
 - `__setitem__`, 68
 - `__str__`, 68
 - `getAllOptions`, 68
 - `perturb`, 69
 - `reset`, 69
 - `val`, 69
 - `val_init`, 69
 - `val_list`, 70
- `skdaccess::framework::param_class::AutoListRemove`
 - `__call__`, 71
 - `__getitem__`, 71
 - `__init__`, 71
 - `__len__`, 72

- __setitem__, 72
 - __str__, 72
 - getAllOptions, 73
 - n, 74
 - perturb, 73
 - reset, 73
 - val, 73
 - val_init, 74
 - val_list, 74
- skdaccess::framework::param_class::AutoListSubset
 - __call__, 75
 - __getitem__, 75
 - __len__, 76
 - __setitem__, 76
 - __str__, 76
 - getAllOptions, 77
 - perturb, 77
 - reset, 77
 - val, 77
 - val_init, 78
 - val_list, 78
- skdaccess::framework::param_class::AutoParam
 - __call__, 80
 - __init__, 79
 - __str__, 80
 - perturb, 80
 - reset, 80
 - val, 81
 - val_init, 81
- skdaccess::framework::param_class::AutoParamList
 - __call__, 82
 - __init__, 82
 - __str__, 82
 - perturb, 83
 - reset, 83
 - val, 83
 - val_init, 83
 - val_list, 83
- skdaccess::framework::param_class::AutoParamListCycle
 - __call__, 85
 - __init__, 85
 - __str__, 85
 - current_index, 86
 - perturb, 85
 - reset, 86
 - val, 86
 - val_init, 86
 - val_list, 86
- skdaccess::framework::param_class::AutoParamMinMax
 - __call__, 88
 - __init__, 87
 - __str__, 88
 - decimals, 89
 - n, 89
 - n_max, 89
 - perturb, 88
 - reset, 89
 - val, 89
 - val_init, 89
 - val_max, 90
 - val_min, 90
- skdaccess::geo::era_interim::cache::data_fetcher::DataFetcher
 - __init__, 196
 - __str__, 196
 - ap_paramList, 200
 - cacheData, 196
 - checkIfDataExists, 197
 - data_names, 200
 - date_list, 201
 - getConfig, 197
 - getDataLocation, 197
 - getHDFStorage, 198
 - getMetadata, 198
 - multirun_enabled, 198
 - output, 199
 - password, 201
 - perturb, 199
 - reset, 199
 - setDataLocation, 199
 - username, 201
 - verbose, 201
 - verbose_print, 200
 - writeConfig, 200
- skdaccess::geo::gldas::data_fetcher::DataFetcher
 - __init__, 91
 - __str__, 92
 - ap_paramList, 95
 - downloadFullDataset, 92
 - end_date, 95
 - getConfig, 93
 - getDataLocation, 93
 - getMetadata, 93
 - multirun_enabled, 94
 - output, 94
 - perturb, 94
 - resample, 96
 - reset, 94
 - setDataLocation, 94
 - start_date, 96
 - verbose, 96
 - verbose_print, 95
 - writeConfig, 95
- skdaccess::geo::grace::data_fetcher::DataFetcher
 - __init__, 234
 - __str__, 235
 - ap_paramList, 238
 - downloadFullDataset, 235

- end_date, 238
- getConfig, 236
- getDataLocation, 236
- getMetadata, 236
- multirun_enabled, 236
- output, 237
- perturb, 237
- reset, 237
- setDataLocation, 237
- start_date, 239
- verbose, 239
- verbose_print, 238
- writeConfig, 238
- skdaccess::geo::grace::mascon::cache::data_fetcher::DataFetcher
 - __init__, 182
 - __str__, 182
 - ap_paramList, 187
 - cacheData, 182
 - checkIfDataExists, 183
 - end_date, 187
 - getConfig, 184
 - getDataLocation, 184
 - getHDFStorage, 184
 - getMasconPlacement, 185
 - getMetadata, 185
 - mascon_placement_url, 187
 - mascon_url, 188
 - multirun_enabled, 185
 - output, 185
 - perturb, 186
 - reset, 186
 - scale_factor_url, 188
 - setDataLocation, 186
 - start_date, 188
 - verbose, 188
 - verbose_print, 187
 - writeConfig, 187
- skdaccess::geo::groundwater::data_fetcher::DataFetcher
 - __init__, 106
 - __str__, 106
 - ap_paramList, 110
 - cutoff, 110
 - downloadFullDataset, 107
 - end_date, 110
 - getConfig, 107
 - getDataLocation, 107
 - getMetadata, 108
 - getStationMetadata, 108
 - multirun_enabled, 108
 - output, 108
 - perturb, 109
 - reset, 109
 - setDataLocation, 109
 - start_date, 111
 - verbose, 111
 - verbose_print, 110
 - writeConfig, 110
- skdaccess::geo::imsdnhs::data_fetcher::DataFetcher
 - __init__, 189
 - __str__, 190
 - ap_paramList, 193
 - coordinate_dict, 193
 - downloadFullDataset, 190
 - end_date, 194
 - getConfig, 191
 - getDataLocation, 191
 - getMetadata, 191
 - multirun_enabled, 191
 - output, 192
 - perturb, 192
 - reset, 192
 - setDataLocation, 192
 - start_date, 194
 - verbose, 194
 - verbose_print, 193
 - writeConfig, 193
- skdaccess::geo::magnetometer::data_fetcher::DataFetcher
 - __init__, 128
 - __str__, 129
 - ap_paramList, 132
 - channels, 132
 - data_type, 132
 - end_time, 132
 - getConfig, 129
 - getDataMetadata, 129
 - getMetadata, 130
 - interval, 133
 - multirun_enabled, 130
 - output, 130
 - perturb, 130
 - reset, 131
 - retrieveOnlineData, 131
 - start_time, 133
 - verbose, 133
 - verbose_print, 131
 - writeConfig, 132
- skdaccess::geo::mahali::rinex::data_fetcher::DataFetcher
 - __init__, 241
 - __str__, 241
 - ap_paramList, 246
 - cacheData, 241
 - checkIfDataExists, 242
 - date_range, 246
 - end_date, 246
 - generate_links, 246
 - getConfig, 243

- getDataLocation, 243
- getHDFStorage, 243
- getMetadata, 244
- multirun_enabled, 244
- output, 244
- perturb, 244
- reset, 245
- setDataLocation, 245
- start_date, 247
- verbose, 247
- verbose_print, 245
- writeConfig, 246
- skdaccess::geo::mahali::rinex::data_wrapper::Data↔
 - Wrapper
 - __len__, 293
 - addResult, 293
 - constants, 296
 - data, 296
 - get, 293
 - getIterator, 294
 - getResults, 294
 - getRunID, 294
 - info, 294
 - meta_data, 296
 - reset, 295
 - results, 296
 - run_id, 296
 - update, 295
 - updateMetadata, 295
- skdaccess::geo::mahali::tec::data_fetcher::DataFetcher
 - __init__, 210
 - __str__, 210
 - ap_paramList, 215
 - cacheData, 211
 - checkIfDataExists, 211
 - date_range, 215
 - end_date, 215
 - getConfig, 212
 - getDataLocation, 212
 - getHDFStorage, 212
 - getMetadata, 213
 - multirun_enabled, 213
 - output, 213
 - perturb, 213
 - reset, 214
 - setDataLocation, 214
 - start_date, 215
 - verbose, 216
 - verbose_print, 214
 - writeConfig, 215
- skdaccess::geo::mahali::temperature::data_fetcher::↔
 - DataFetcher
 - __init__, 248
 - __str__, 249
- ap_paramList, 251
- end_date, 252
- getConfig, 249
- getMetadata, 249
- multirun_enabled, 249
- output, 250
- perturb, 250
- reset, 250
- retrieveOnlineData, 250
- start_date, 252
- verbose, 252
- verbose_print, 251
- writeConfig, 251
- skdaccess::geo::modis::cache::cloud_mask::data_↔
 - fetcher::DataFetcher
 - __init__, 150
- skdaccess::geo::modis::cache::cloud_opacity::data_↔
 - fetcher::DataFetcher
 - __init__, 149
- skdaccess::geo::modis::cache::data_fetcher::DataFetcher
 - __init__, 154
 - __str__, 155
 - ap_paramList, 160
 - cacheData, 155
 - checkIfDataExists, 156
 - daynightboth, 160
 - end_date, 161
 - find_data, 157
 - getConfig, 157
 - getDataLocation, 157
 - getHDFStorage, 158
 - getMetadata, 158
 - grid, 161
 - grid_fill, 161
 - modis_id, 161
 - modis_identifier, 161
 - modis_platform, 161
 - multirun_enabled, 158
 - output, 159
 - perturb, 159
 - reset, 159
 - setDataLocation, 159
 - start_date, 161
 - use_long_name, 162
 - variable_list, 162
 - verbose, 162
 - verbose_print, 160
 - writeConfig, 160
- skdaccess::geo::modis::cache::reflectance::data_↔
 - fetcher::DataFetcher
 - __init__, 152
- skdaccess::geo::modis::stream::cloud_mask::data_↔
 - fetcher::DataFetcher
 - __init__, 164

- skdaccess::geo::modis::stream::cloud_opacity::data_↵
 fetcher::DataFetcher
 __init__, 163
- skdaccess::geo::modis::stream::data_fetcher::Data_↵
 Fetcher
 __init__, 175
 __str__, 176
 ap_paramList, 178
 daynightboth, 179
 end_date, 179
 getConfig, 176
 getMetadata, 176
 grid, 179
 grid_fill, 179
 modis_id, 179
 modis_identifier, 179
 modis_platform, 179
 multirun_enabled, 176
 output, 177
 perturb, 177
 reset, 177
 retrieveOnlineData, 177
 start_date, 180
 use_long_name, 180
 variable_list, 180
 verbose, 180
 verbose_print, 178
 writeConfig, 178
- skdaccess::geo::modis::stream::reflectance::data_↵
 fetcher::DataFetcher
 __init__, 134
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher
 __init__, 203
 __str__, 203
 ap_paramList, 207
 data_type, 207
 downloadFullDataset, 203
 end_date, 207
 getAntennaLogs, 204
 getConfig, 204
 getDataLocation, 204
 getMetadata, 205
 getStationMetadata, 205
 lat_range, 208
 lon_range, 208
 mdyratio, 208
 multirun_enabled, 205
 output, 205
 perturb, 206
 reset, 206
 setDataLocation, 206
 start_date, 208
 verbose, 208
 verbose_print, 207
- writeConfig, 207
- skdaccess::geo::pbo::data_fetcher::DataFetcher
 __init__, 226
 __str__, 227
 antenna_info, 232
 ap_paramList, 232
 default_columns, 232
 default_error_columns, 232
 downloadFullDataset, 227
 getAntennaLogs, 228
 getConfig, 228
 getDataLocation, 228
 getInfo, 228
 getMetadata, 229
 getStationMetadata, 229
 index_date_only, 232
 meta_data, 232
 multirun_enabled, 229
 output, 229
 perturb, 230
 reset, 230
 setDataLocation, 230
 setStationList, 231
 station_list, 232
 use_progress_bar, 233
 verbose, 233
 verbose_print, 231
 writeConfig, 231
- skdaccess::geo::sentinel_1::cache::data_fetcher::Data_↵
 Fetcher
 __init__, 98
 __str__, 98
 ap_paramList, 103
 cacheData, 98
 checkIfDataExists, 99
 getConfig, 100
 getDataLocation, 100
 getHDFStorage, 100
 getMetadata, 101
 local_paths, 103
 multirun_enabled, 101
 output, 101
 password, 103
 perturb, 101
 polarization, 103
 reset, 102
 satellite_url_list, 104
 setDataLocation, 102
 swath, 104
 url_list, 104
 username, 104
 verbose, 104
 verbose_print, 102
 writeConfig, 103

skdaccess::geo::srtm::cache::data_fetcher::DataFetcher

- __init__, 113
- __str__, 113
- ap_paramList, 118
- arcsecond_sampling, 118
- cacheData, 114
- checkIfDataExists, 114
- getConfig, 115
- getDataLocation, 115
- getHDFStorage, 115
- getMetadata, 116
- lat_tile_end, 118
- lat_tile_start, 119
- lon_tile_end, 119
- lon_tile_start, 119
- mask_water, 119
- multirun_enabled, 116
- output, 116
- password, 119
- perturb, 116
- reset, 117
- setDataLocation, 117
- store_geolocation_grids, 119
- username, 119
- verbose, 120
- verbose_print, 117
- writeConfig, 118

skdaccess::geo::uavsar::cache::data_fetcher::Data↵
Fetcher

- __init__, 121
- __str__, 122
- ap_paramList, 126
- cacheData, 122
- checkIfDataExists, 123
- getConfig, 123
- getDataLocation, 123
- getHDFStorage, 124
- getMetadata, 124
- llh_url, 126
- memmap, 127
- metadata_url_list, 127
- multirun_enabled, 124
- output, 125
- perturb, 125
- reset, 125
- setDataLocation, 125
- slc_url_list, 127
- verbose, 127
- verbose_print, 126
- writeConfig, 126

skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher

- __init__, 136
- __str__, 137

- ap_paramList, 141
- cacheData, 137
- checkIfDataExists, 137
- day_end, 141
- day_start, 141
- end_hour, 142
- getConfig, 138
- getDataLocation, 138
- getHDFStorage, 138
- getMetadata, 139
- month_list, 142
- multirun_enabled, 139
- output, 139
- perturb, 140
- reset, 140
- setDataLocation, 140
- start_hour, 142
- station_number, 142
- verbose, 142
- verbose_print, 141
- writeConfig, 141
- year_list, 142

skdaccess::geo::wyoming_sounding::stream::data_↵
fetcher::DataFetcher

- __init__, 144
- __str__, 145
- ap_paramList, 147
- day_end, 148
- day_start, 148
- end_hour, 148
- getConfig, 145
- getMetadata, 145
- month_list, 148
- multirun_enabled, 145
- output, 145, 146
- perturb, 146
- reset, 146
- retrieveOnlineData, 146
- start_hour, 148
- station_number, 148
- verbose, 148
- verbose_print, 147
- writeConfig, 147
- year_list, 149

skdaccess::planetary::ode::cache::data_fetcher::Data↵
Fetcher

- __init__, 166
- __str__, 167
- ap_paramList, 171
- cacheData, 167
- checkIfDataExists, 167
- eastern_lon, 171
- file_name, 171
- getConfig, 168

- getDataLocation, 168
- getHDFStorage, 168
- getMetadata, 169
- instrument, 172
- max_lat, 172
- max_ob_time, 172
- min_lat, 172
- min_ob_time, 172
- mission, 172
- multirun_enabled, 169
- number_product_limit, 172
- output, 169
- perturb, 170
- product_id, 173
- product_type, 173
- remove_ndv, 173
- reset, 170
- result_offset_number, 173
- setDataLocation, 170
- target, 173
- verbose, 173
- verbose_print, 170
- western_lon, 173
- writeConfig, 171
- skdaccess::solar::sdo::data_fetcher::DataFetcher
 - __init__, 263
 - __str__, 263
 - ap_paramList, 266
 - getConfig, 263
 - getMetadata, 264
 - multirun_enabled, 264
 - output, 264
 - perturb, 264
 - reset, 265
 - retrieveOnlineData, 265
 - verbose, 266
 - verbose_print, 265
 - writeConfig, 266
- skdaccess::utilities::file_browser::FileBrowser
 - __init__, 302
 - dirs, 303
 - files, 303
 - path, 303
 - widget, 303
- skdaccess::utilities::grace_util
 - averageDates, 27
 - computeEWD, 28
 - dateMismatch, 28
 - getStartEndDate, 30
 - readTellusData, 30
- skdaccess::utilities::gw_util
 - combine_water_heights, 31
- skdaccess::utilities::image_util
 - convertBinCentersToEdges, 32
 - getExtentsFromCentersPlateCarree, 32
 - getGeoTransform, 33
 - lat_spline, 34
 - lon_spline, 34
 - SplineGeolocation, 33
 - x_offset, 34
 - x_spline, 34
 - y_offset, 34
 - y_spline, 34
- skdaccess::utilities::image_util::AffineGlobalCoords
 - __init__, 55
 - getPixelYX, 56
 - getProjectedYX, 56
- skdaccess::utilities::image_util::LinearGeolocation
 - __init__, 313
 - flip_y, 314
 - getExtents, 313
 - getLatLon, 313
 - getYX, 314
 - lat_extents, 314
 - lat_pixel_size, 314
 - len_x, 315
 - len_y, 315
 - lon_extents, 315
 - lon_pixel_size, 315
 - start_lat, 315
 - start_lon, 315
 - x_offset, 315
 - y_offset, 316
- skdaccess::utilities::image_util::SplineLatLon
 - __call__, 333
 - __init__, 332
 - lat_func, 334
 - lon_func, 334
 - x_offset, 334
 - y_offset, 334
- skdaccess::utilities::kepler_util
 - normalize, 35
- skdaccess::utilities::mahali_util
 - convert_date, 35
 - parselonoFile, 36
- skdaccess::utilities::modis_util
 - calibrateModis, 36
 - checkBit, 37
 - createGrid, 37
 - getFileIDs, 38
 - getFileURLs, 39
 - getImageType, 39
 - getModisData, 40
 - readMODISData, 40
 - rescale, 40
- skdaccess::utilities::modis_util::LatLon
 - __call__, 310
 - __init__, 310

- alat, [311](#)
- alon, [311](#)
- lat_data, [311](#)
- lon_data, [311](#)
- x_offset, [311](#)
- y_offset, [311](#)
- skdaccess::utilities::ode_util
 - correct_CRISM_label, [41](#)
 - correct_file_name_case_in_label, [42](#)
 - correct_label_file, [42](#)
 - get_files_urls, [42](#)
 - get_query_url, [42](#)
 - get_raster_array, [43](#)
 - get_raster_extent, [43](#)
 - query_files_urls, [44](#)
 - query_yes_no, [45](#)
- skdaccess::utilities::pbo_util
 - getLatLonRange, [45](#)
 - getROIstations, [46](#)
 - getStationCoords, [46](#)
 - nostab_sys, [47](#)
 - propagateErrors, [47](#)
 - removeAntennaOffset, [48](#)
 - stab_sys, [48](#)
- skdaccess::utilities::sentinel_1_util
 - parseSatelliteData, [49](#)
- skdaccess::utilities::sounding_util
 - generateQueries, [50](#)
- skdaccess::utilities::sounding_util::SoundingParser
 - __init__, [329](#)
 - data_dict, [330](#)
 - handle_data, [329](#)
 - handle_endtag, [330](#)
 - handle_starttag, [330](#)
 - in_header, [330](#)
 - in_pre_tag, [331](#)
 - label, [331](#)
 - metadata_dict, [331](#)
 - read_data, [331](#)
 - tmp, [331](#)
- skdaccess::utilities::srtm_util
 - getSRTMData, [51](#)
 - getSRTMLatLon, [51](#)
 - merge_srtm_tiles, [52](#)
- skdaccess::utilities::support
 - convertToStr, [52](#)
 - progress_bar, [53](#)
 - retrieveCommonDatesHDF, [53](#)
- skdaccess::utilities::uavsar_util
 - readUAVSARMetadata, [54](#)
- slc_url_list
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔
Fetcher, [127](#)
- solar/sdo/data_fetcher.py, [350](#)
- spacecraft_list
 - skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [261](#)
- SplineGeolocation
 - skdaccess::utilities::image_util, [33](#)
- stab_sys
 - skdaccess::utilities::pbo_util, [48](#)
- start_date
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, [96](#)
 - skdaccess::geo::grace::data_fetcher::DataFetcher, [239](#)
 - skdaccess::geo::grace::mascon::cache::data↔
fetcher::DataFetcher, [188](#)
 - skdaccess::geo::groundwater::data_fetcher::Data↔
Fetcher, [111](#)
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔
Fetcher, [194](#)
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔
Fetcher, [247](#)
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔
Fetcher, [215](#)
 - skdaccess::geo::mahali::temperature::data_fetcher↔
::DataFetcher, [252](#)
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, [161](#)
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, [180](#)
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, [208](#)
- start_hour
 - skdaccess::geo::wyoming_sounding::cache::data↔
fetcher::DataFetcher, [142](#)
 - skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, [148](#)
- start_lat
 - skdaccess::utilities::image_util::LinearGeolocation, [315](#)
- start_lon
 - skdaccess::utilities::image_util::LinearGeolocation, [315](#)
- start_time
 - skdaccess::geo::magnetometer::data_fetcher::↔
DataFetcher, [133](#)
- station_list
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, [232](#)
- station_number
 - skdaccess::geo::wyoming_sounding::cache::data↔
fetcher::DataFetcher, [142](#)
 - skdaccess::geo::wyoming_sounding::stream::data↔
_fetcher::DataFetcher, [148](#)
- store_geolocation_grids
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, [119](#)

- swath
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 104
- target
 - skdaccess::planetary::ode::cache::data_fetcher::↔
DataFetcher, 173
- tmp
 - skdaccess::utilities::sounding_util::SoundingParser,
331
- update
 - skdaccess::framework::data_class::DataWrapper↔
Base, 300
 - skdaccess::framework::data_class::ImageWrapper,
307
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 320
 - skdaccess::framework::data_class::SeriesWrapper,
326
 - skdaccess::framework::data_class::TableWrapper,
340
 - skdaccess::framework::data_class::XArrayWrapper,
346
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 295
- updateData
 - skdaccess::framework::data_class::ImageWrapper,
307
 - skdaccess::framework::data_class::TableWrapper,
341
- updateFrames
 - skdaccess::framework::data_class::TableWrapper,
341
- updateMetadata
 - skdaccess::framework::data_class::DataWrapper↔
Base, 301
 - skdaccess::framework::data_class::ImageWrapper,
308
 - skdaccess::framework::data_class::SeriesDictionary↔
Wrapper, 320
 - skdaccess::framework::data_class::SeriesWrapper,
327
 - skdaccess::framework::data_class::TableWrapper,
341
 - skdaccess::framework::data_class::XArrayWrapper,
347
 - skdaccess::geo::mahali::rinex::data_wrapper::Data↔
Wrapper, 295
- url_list
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 104
- use_long_name
 - skdaccess::geo::modis::cache::data_fetcher::Data↔
Fetcher, 162
 - skdaccess::geo::modis::stream::data_fetcher::↔
DataFetcher, 180
- use_progress_bar
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 233
- username
 - skdaccess::geo::era_interim::cache::data_fetcher::↔
DataFetcher, 201
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔
DataFetcher, 104
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔
Fetcher, 119
- utilities/file_browser.py, 359
- utilities/grace_util.py, 359
- utilities/gw_util.py, 359
- utilities/image_util.py, 360
- utilities/kepler_util.py, 360
- utilities/mahali_util.py, 361
- utilities/modis_util.py, 361
- utilities/ode_util.py, 362
- utilities/pbo_util.py, 362
- utilities/sentinel_1_util.py, 363
- utilities/sounding_util.py, 363
- utilities/srtm_util.py, 363
- utilities/support.py, 364
- utilities/uavsar_util.py, 364
- val
 - skdaccess::framework::param_class::AutoList, 60
 - skdaccess::framework::param_class::AutoListCycle,
65
 - skdaccess::framework::param_class::AutoList↔
Permute, 69
 - skdaccess::framework::param_class::AutoList↔
Remove, 73
 - skdaccess::framework::param_class::AutoList↔
Subset, 77
 - skdaccess::framework::param_class::AutoParam, 81
 - skdaccess::framework::param_class::AutoParamList,
83
 - skdaccess::framework::param_class::AutoParam↔
ListCycle, 86
 - skdaccess::framework::param_class::AutoParam↔
MinMax, 89
- val_init
 - skdaccess::framework::param_class::AutoList, 61
 - skdaccess::framework::param_class::AutoListCycle,
65
 - skdaccess::framework::param_class::AutoList↔
Permute, 69
 - skdaccess::framework::param_class::AutoList↔
Remove, 74
 - skdaccess::framework::param_class::AutoList↔
Subset, 78
 - skdaccess::framework::param_class::AutoParam, 81

- skdaccess::framework::param_class::AutoParamList, 83
- skdaccess::framework::param_class::AutoParam↔ListCycle, 86
- skdaccess::framework::param_class::AutoParam↔MinMax, 89
- val_list
 - skdaccess::framework::param_class::AutoList, 61
 - skdaccess::framework::param_class::AutoListCycle, 66
 - skdaccess::framework::param_class::AutoList↔Permute, 70
 - skdaccess::framework::param_class::AutoList↔Remove, 74
 - skdaccess::framework::param_class::AutoList↔Subset, 78
 - skdaccess::framework::param_class::AutoParamList, 83
 - skdaccess::framework::param_class::AutoParam↔ListCycle, 86
- val_max
 - skdaccess::framework::param_class::AutoParam↔MinMax, 90
- val_min
 - skdaccess::framework::param_class::AutoParam↔MinMax, 90
- variable_list
 - skdaccess::geo::modis::cache::data_fetcher::Data↔Fetcher, 162
 - skdaccess::geo::modis::stream::data_fetcher::↔DataFetcher, 180
- verbose
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 224
 - skdaccess::astro::voyager::data_fetcher::Data↔Fetcher, 261
 - skdaccess::framework::data_class::DataFetcher↔Base, 270
 - skdaccess::framework::data_class::DataFetcher↔Cache, 277
 - skdaccess::framework::data_class::DataFetcher↔Local, 282
 - skdaccess::framework::data_class::DataFetcher↔Storage, 287
 - skdaccess::framework::data_class::DataFetcher↔Stream, 291
 - skdaccess::geo::era_interim::cache::data_fetcher::↔DataFetcher, 201
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 96
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 239
 - skdaccess::geo::grace::mascon::cache::data_↔fetcher::DataFetcher, 188
 - skdaccess::geo::groundwater::data_fetcher::Data↔Fetcher, 111
 - skdaccess::geo::imsdnhs::data_fetcher::Data↔Fetcher, 194
 - skdaccess::geo::magnetometer::data_fetcher::↔DataFetcher, 133
 - skdaccess::geo::mahali::rinex::data_fetcher::Data↔Fetcher, 247
 - skdaccess::geo::mahali::tec::data_fetcher::Data↔Fetcher, 216
 - skdaccess::geo::mahali::temperature::data_fetcher↔::DataFetcher, 252
 - skdaccess::geo::modis::cache::data_fetcher::Data↔Fetcher, 162
 - skdaccess::geo::modis::stream::data_fetcher::↔DataFetcher, 180
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 208
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 233
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↔DataFetcher, 104
 - skdaccess::geo::srtm::cache::data_fetcher::Data↔Fetcher, 120
 - skdaccess::geo::uavsar::cache::data_fetcher::Data↔Fetcher, 127
 - skdaccess::geo::wyoming_sounding::cache::data_↔fetcher::DataFetcher, 142
 - skdaccess::geo::wyoming_sounding::stream::data_↔fetcher::DataFetcher, 148
 - skdaccess::planetary::ode::cache::data_fetcher::↔DataFetcher, 173
 - skdaccess::solar::sdo::data_fetcher::DataFetcher, 266
- verbose_print
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 223
 - skdaccess::astro::voyager::data_fetcher::Data↔Fetcher, 260
 - skdaccess::framework::data_class::DataFetcher↔Base, 270
 - skdaccess::framework::data_class::DataFetcher↔Cache, 276
 - skdaccess::framework::data_class::DataFetcher↔Local, 281
 - skdaccess::framework::data_class::DataFetcher↔Storage, 286
 - skdaccess::framework::data_class::DataFetcher↔Stream, 290
 - skdaccess::geo::era_interim::cache::data_fetcher::↔DataFetcher, 200
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 95
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 238

- skdaccess::geo::grace::mascon::cache::data_↵
fetcher::DataFetcher, 187
- skdaccess::geo::groundwater::data_fetcher::Data_↵
Fetcher, 110
- skdaccess::geo::imsdnhs::data_fetcher::Data_↵
Fetcher, 193
- skdaccess::geo::magnetometer::data_fetcher::↵
DataFetcher, 131
- skdaccess::geo::mahali::rinex::data_fetcher::Data_↵
Fetcher, 245
- skdaccess::geo::mahali::tec::data_fetcher::Data_↵
Fetcher, 214
- skdaccess::geo::mahali::temperature::data_fetcher_↵
::DataFetcher, 251
- skdaccess::geo::modis::cache::data_fetcher::Data_↵
Fetcher, 160
- skdaccess::geo::modis::stream::data_fetcher::↵
DataFetcher, 178
- skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 207
- skdaccess::geo::pbo::data_fetcher::DataFetcher, 231
- skdaccess::geo::sentinel_1::cache::data_fetcher::↵
DataFetcher, 102
- skdaccess::geo::srtm::cache::data_fetcher::Data_↵
Fetcher, 117
- skdaccess::geo::uavsar::cache::data_fetcher::Data_↵
Fetcher, 126
- skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 141
- skdaccess::geo::wyoming_sounding::stream::data_↵
_fetcher::DataFetcher, 147
- skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 170
- skdaccess::solar::sdo::data_fetcher::DataFetcher, 265
- western_lon
 - skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 173
- widget
 - skdaccess::utilities::file_browser::FileBrowser, 303
- writeConfig
 - skdaccess::astro::kepler::data_fetcher::DataFetcher, 224
 - skdaccess::astro::voyager::data_fetcher::Data_↵
Fetcher, 260
 - skdaccess::framework::data_class::DataFetcher_↵
Base, 270
 - skdaccess::framework::data_class::DataFetcher_↵
Cache, 277
 - skdaccess::framework::data_class::DataFetcher_↵
Local, 282
 - skdaccess::framework::data_class::DataFetcher_↵
Storage, 287
 - skdaccess::framework::data_class::DataFetcher_↵
Stream, 291
 - skdaccess::geo::era_interim::cache::data_fetcher::↵
DataFetcher, 200
 - skdaccess::geo::gldas::data_fetcher::DataFetcher, 95
 - skdaccess::geo::grace::data_fetcher::DataFetcher, 238
 - skdaccess::geo::grace::mascon::cache::data_↵
fetcher::DataFetcher, 187
 - skdaccess::geo::groundwater::data_fetcher::Data_↵
Fetcher, 110
 - skdaccess::geo::imsdnhs::data_fetcher::Data_↵
Fetcher, 193
 - skdaccess::geo::magnetometer::data_fetcher::↵
DataFetcher, 132
 - skdaccess::geo::mahali::rinex::data_fetcher::Data_↵
Fetcher, 246
 - skdaccess::geo::mahali::tec::data_fetcher::Data_↵
Fetcher, 215
 - skdaccess::geo::mahali::temperature::data_fetcher_↵
::DataFetcher, 251
 - skdaccess::geo::modis::cache::data_fetcher::Data_↵
Fetcher, 160
 - skdaccess::geo::modis::stream::data_fetcher::↵
DataFetcher, 178
 - skdaccess::geo::ngl_gps::data_fetcher::DataFetcher, 207
 - skdaccess::geo::pbo::data_fetcher::DataFetcher, 231
 - skdaccess::geo::sentinel_1::cache::data_fetcher::↵
DataFetcher, 103
 - skdaccess::geo::srtm::cache::data_fetcher::Data_↵
Fetcher, 118
 - skdaccess::geo::uavsar::cache::data_fetcher::Data_↵
Fetcher, 126
 - skdaccess::geo::wyoming_sounding::cache::data_↵
fetcher::DataFetcher, 141
 - skdaccess::geo::wyoming_sounding::stream::data_↵
_fetcher::DataFetcher, 147
 - skdaccess::planetary::ode::cache::data_fetcher::↵
DataFetcher, 171
 - skdaccess::solar::sdo::data_fetcher::DataFetcher, 266
- x_offset
 - skdaccess::utilities::image_util, 34
 - skdaccess::utilities::image_util::LinearGeolocation, 315
 - skdaccess::utilities::image_util::SplineLatLon, 334
 - skdaccess::utilities::modis_util::LatLon, 311
- x_spline
 - skdaccess::utilities::image_util, 34
- y_offset
 - skdaccess::utilities::image_util, 34

skdaccess::utilities::image_util::LinearGeolocation,
[316](#)

skdaccess::utilities::image_util::SplineLatLon, [334](#)

skdaccess::utilities::modis_util::LatLon, [311](#)

y_spline

skdaccess::utilities::image_util, [34](#)

year_list

skdaccess::astro::voyager::data_fetcher::Data↔
Fetcher, [261](#)

skdaccess::geo::wyoming_sounding::cache::data_↔
fetcher::DataFetcher, [142](#)

skdaccess::geo::wyoming_sounding::stream::data_↔
_fetcher::DataFetcher, [149](#)