

CMIP6 Data Request

Computational Implementation
and
Workflow

Martin Schupfner
Deutsches Klimarechenzentrum (DKRZ)

Outline

- CMIP6 Data Request
 - Structure of the Data Request
 - Data Request Python API (DreqPy)
 - Data Request Web GUI
- Workflow
 - Overview
 - Data Request
- Variable Mapping Modell ↔ CMOR
 - Post-Processing
 - MIP Table ↔ Model
 - Data Request Web GUI

CMIP6 Data Request

Which variables shall be provided?

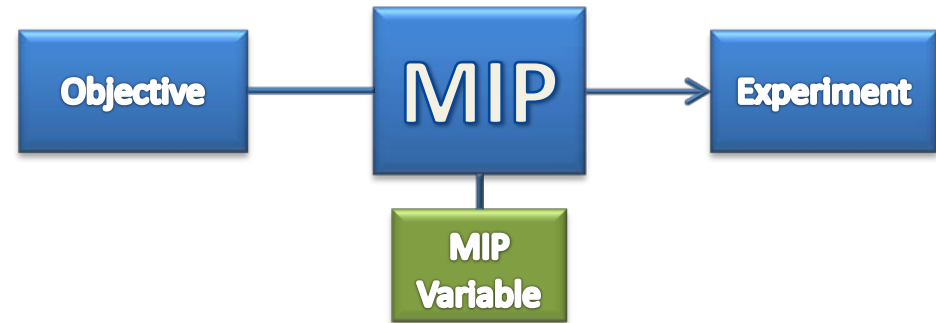
CMIP6 Data Request Structure

- MIPs founded to achieve scientific objectives



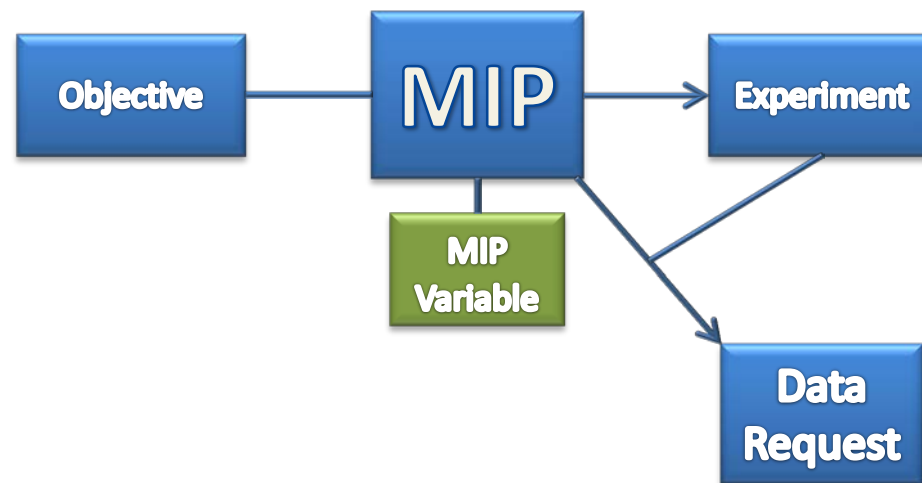
CMIP6 Data Request Structure

- MIPs founded to achieve scientific objectives
- MIPs define Experiments, Variables



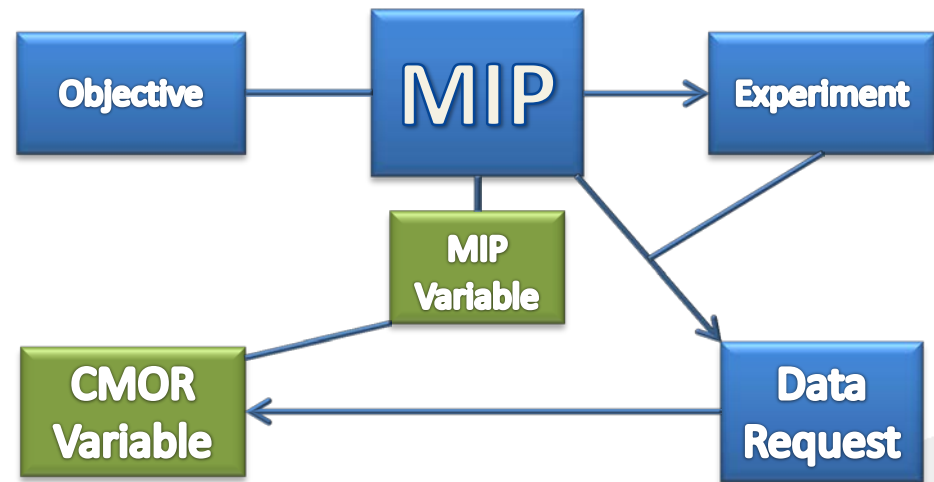
CMIP6 Data Request Structure

- MIPs founded to achieve scientific objectives
- MIPs define Experiments, Variables and set up a data request



CMIP6 Data Request Structure

- MIPs founded to achieve scientific objectives
- MIPs define Experiments, Variables and set up a data request
- CMOR-Variables are the different realisations (frequency, shape, ...) of a MIP-Variable



Example:

MIP-Variable: Ozone volume mixing ratio

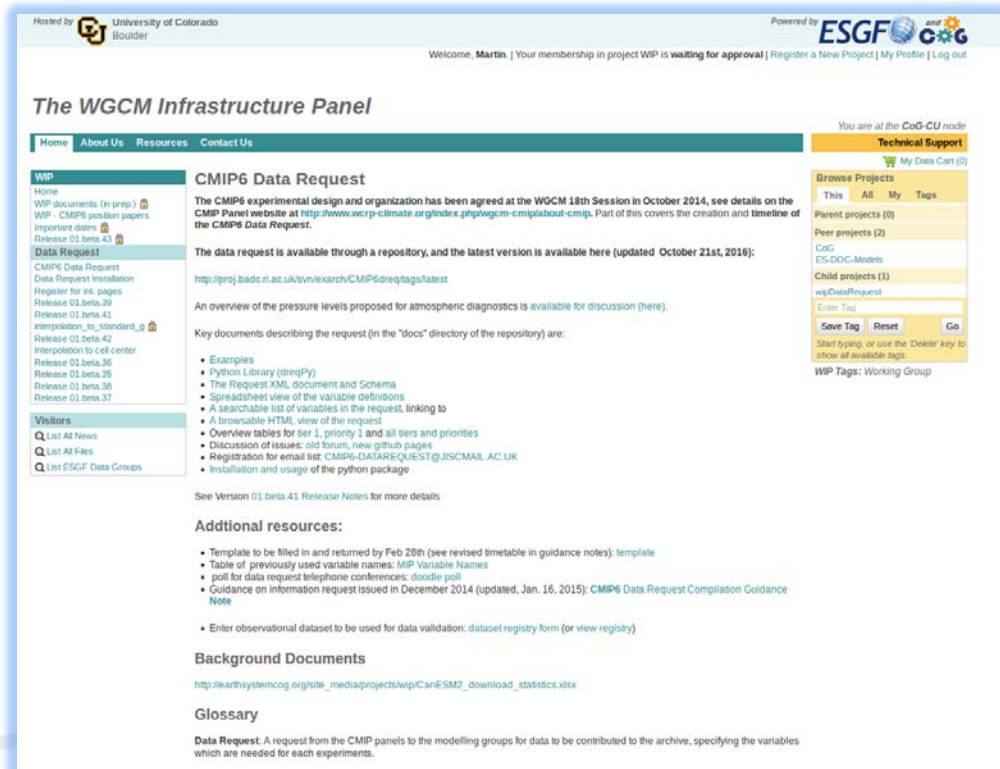
CMOR Variables:

- (1) Ozone vmr (zonal mean on 39 pressure levels, monthly mean)
- (2) Ozone vmr (global field on model levels, monthly mean)
- (3) Ozone vmr (global field on 23 pressure levels, monthly mean)

CMIP6 Data Request Publication

The Data Request is published as:

- 2 xml-files
- DreqPy API (Python Tool)
- All-In-One Excel Sheet (created using DreqPy)
- Coordinated by WIP (WGCM Infrastructure Panel)

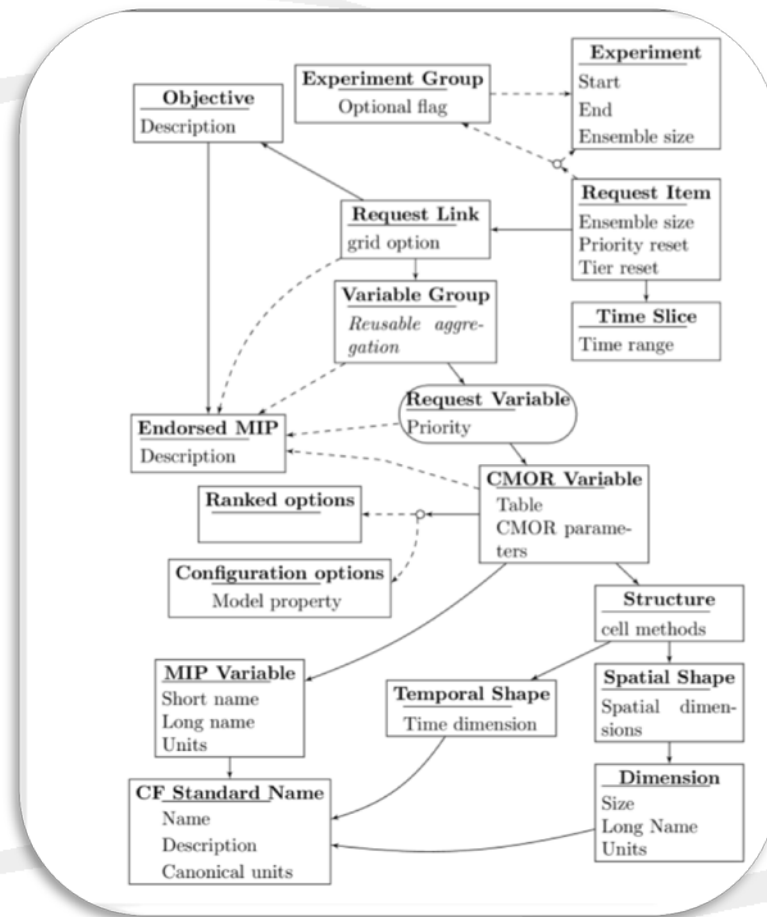


The screenshot shows the 'The WGCM Infrastructure Panel' website. The main heading is 'CMIP6 Data Request'. The text on the page states: 'The CMIP6 experimental design and organization has been agreed at the WGCM 18th Session in October 2014, see details on the CMIP Panel website at <http://www.wgp-climate.org/index.php/wgcm-cmip/about-cmip>. Part of this covers the creation and timeline of the CMIP6 Data Request.' It also mentions that the data request is available through a repository and provides a link to the latest version (updated October 21st, 2016): <http://proj.badc.rl.ac.uk/view/xarch/CMIP6dreq/tags/latest>. A sidebar on the left contains navigation links like 'Home', 'About Us', 'Resources', and 'Contact Us'. A right sidebar shows 'Browse Projects' and 'Technical Support'.

<https://earthsystemcog.org/projects/wip/CMIP6DataRequest>

Data Request Python API (DreqPy)

By Martin Jukes, BADC (2016)



Martin Jukes, BADC (2016)

Installation

Installation from the **SVN** Repository:

```
svn co http://proj.badc.rl.ac.uk/svn/exarch/CMIP6dreq/tags/latest  
python setup.py build  
python setup.py install
```

Prerequisites:

xlsxwriter, Python in Version 2.6.6, 2.7 oder 3.x

Installation: `pysetuptools + svn`

Commandline Interface

```
drq -m ScenarioMIP,CMIP -t 2 -p 1
```

Computes a volume estimate

```
k204212@k204212-ThinkPad-MS:~$ drq -m ScenarioMIP,CMIP -t 2 -p 1
CMIP:: Frequency: 3hr: 1.618932096
CMIP:: Frequency: 6hr: 6.695597376
CMIP:: Frequency: day: 3.585832416
CMIP:: Frequency: dec: 0.021132811056
CMIP:: Frequency: fx: 0.0003929812824
CMIP:: Frequency: mon: 7.20976829616
CMIP:: Frequency: monClim: 0.001492992
CMIP:: Frequency: yr: 0.511415856
*TOTAL:: Frequency: 3hr: 1.618932096
*TOTAL:: Frequency: 6hr: 6.695597376
*TOTAL:: Frequency: day: 3.585832416
*TOTAL:: Frequency: dec: 0.021132811056
*TOTAL:: Frequency: fx: 0.0003929812824
*TOTAL:: Frequency: mon: 7.20976829616
*TOTAL:: Frequency: monClim: 0.001492992
*TOTAL:: Frequency: yr: 0.511415856
CMIP:: TOTAL volume: 19.64Tb
ScenarioMIP:: TOTAL volume: 0.00Tb
*TOTAL:: TOTAL volume: 19.64Tb
```

Commandline Interface

drq
Com

```
k204212@k204212-ThinkPad-MS:~$ drq -m ScenarioMIP,CMIP -t 2 -p 1
CMIP:: Frequency: 3hr: 1.618932096
CMIP:: Frequency: 6hr: 6.695597376
CMIP:: Frequency: day: 3.585832416
CMIP:: Frequency: dec: 0.021132811056
CMIP:: Frequency: fx: 0.0003929812824
CMIP:: Frequency: mon: 7.20976829616
CMIP:: Frequency: monClim: 0.001492992
CMIP:: Frequency: yr: 0.511415856
*TOTAL:: Frequency: 3hr: 1.618932096
*TOTAL:: Frequency: 6hr: 6.695597376
*TOTAL:: Frequency: day: 3.585832416
*TOTAL:: Frequency: dec: 0.021132811056
*TOTAL:: Frequency: fx: 0.0003929812824
*TOTAL:: Frequency: mon: 7.20976829616
*TOTAL:: Frequency: monClim: 0.001492992
*TOTAL:: Frequency: yr: 0.511415856
CMIP:: TOTAL volume: 19.64Tb
ScenarioMIP:: TOTAL volume: 0.00Tb
*TOTAL:: TOTAL volume: 19.64Tb
```

Commandline Interface

```
drq -m ScenarioMIP,CMIP -t 2 -p 1
```

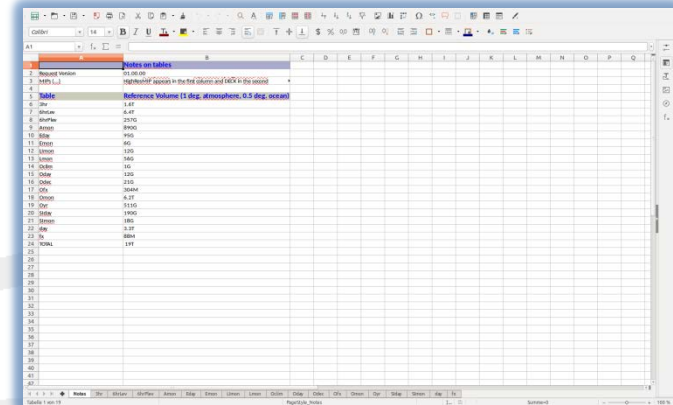
Computes a volume estimate

```
drq -m ScenarioMIP,CMIP -t 2 -p 1
```

```
--xls
```

Additionally creates MS-Excel-Sheet

```
k204212@k204212-ThinkPad-MS:~$ drq -m ScenarioMIP,CMIP -t 2 -p 1
CMIP:: Frequency: 3hr: 1.618932096
CMIP:: Frequency: 6hr: 6.695597376
CMIP:: Frequency: day: 3.585832416
CMIP:: Frequency: dec: 0.021132811056
CMIP:: Frequency: fx: 0.0003929812824
CMIP:: Frequency: mon: 7.20976829616
CMIP:: Frequency: monClim: 0.001492992
CMIP:: Frequency: yr: 0.511415856
*TOTAL:: Frequency: 3hr: 1.618932096
*TOTAL:: Frequency: 6hr: 6.695597376
*TOTAL:: Frequency: day: 3.585832416
*TOTAL:: Frequency: dec: 0.021132811056
*TOTAL:: Frequency: fx: 0.0003929812824
*TOTAL:: Frequency: mon: 7.20976829616
*TOTAL:: Frequency: monClim: 0.001492992
*TOTAL:: Frequency: yr: 0.511415856
CMIP:: TOTAL volume: 19.64Tb
ScenarioMIP:: TOTAL volume: 0.00Tb
*TOTAL:: TOTAL volume: 19.64Tb
```

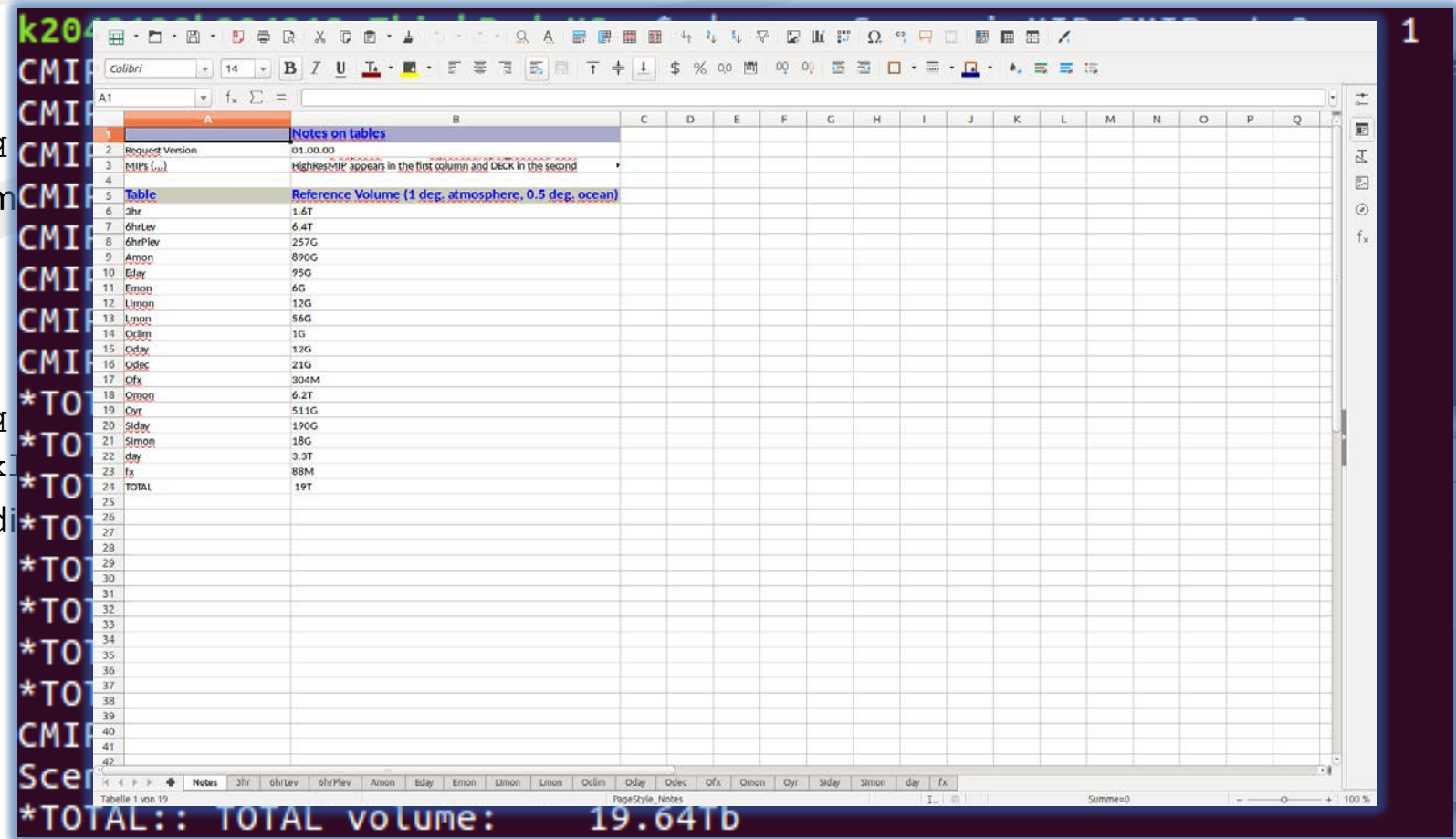


Frequency	Volume
3hr	1.618932096
6hr	6.695597376
day	3.585832416
dec	0.021132811056
fx	0.0003929812824
mon	7.20976829616
monClim	0.001492992
yr	0.511415856
TOTAL	19.64Tb
ScenarioMIP	0.00Tb
TOTAL	19.64Tb

Commandline Interface

drq
Com
drq
--x
Add

1



Request Version	Notes on tables
01.00.00	
	HighResMIP appears in the first column and DECK in the second
Table	Reference Volume (1 deg. atmosphere, 0.5 deg. ocean)
3hr	1.4T
6hrLev	6.4T
6hrPlev	257G
Amon	890G
Eday	95G
Emon	6G
Limon	12G
Lmon	56G
Oclim	1G
Oday	12G
Odec	21G
Ofx	304M
Omon	6.2T
Oyr	511G
Siday	190G
Simon	18G
day	3.3T
fx	88M
TOTAL	19T

Data Structure

Content-Object: dq.coll

- Python dictionary: Dict[key]=value
- Contains all sections of the DataRequest (eg. MIPs, Grids, Variables, ...)
- For each section the following information is available:
 - header - named tuple with title, label, etc. of each section
 - attDefn - Dictionary, contains dataset attributes of each section
 - items - List of records of each section

Index-Object: dq.inx

- Look-up table
 - uid - Search item via UID
 - iref - Search for references of an item in all sections
 - iref.bysect - Search for references of an item in a specific section

Further Information

Official Links, examples & further Information:

- Redmine Project-Wiki
<https://redmine.dkrz.de/projects/cmip6-dicad/wiki>
- Soon: <https://c6dreq.dkrz.de>
- Ipython Notebook

DreqPy (Data Request Python API)

Basic Imports

```
In [ ]: #!/usr/bin/python
# -*- coding: utf-8 -*-

# IPython Notebook created by Martin Schupfner, DKRZ
# Reference: Martin Juckes 2016 -
#           dreqPy (Data Request Python API) User's Guide

from dreqPy import dreq, scope

print "Using DreqPy (Data Request Python API) in version %s" \
      % str(dreq.version)

# Initialisation
dq = dreq.loadDreq()
```

dq.coll Examples

```
In [ ]: # dq.coll
# Content Object dq.coll is a dictionary containing the data request sections,
#   with 3 elements (named tuples) for each section:
# - header : named tuple with info such as title, label, etc.
# - attDefn: dictionary containing record attribute definitions
# - items  : list of records

# Print all entries of dq.coll
print "dq.coll Entries:\n", ", ".join(dq.coll.keys())
```


DreqPy API

by Martin Juckes, BADC

- Interface for the CMIP6 Data Request written in Python
- Build customized Data Request (depending on MIP, Experiment, Variable Priority, Experiment Tier)
- Interactive Browsing of the Data Request possible via a collection of classes and functions
- Calculate Data Volume Estimates
- Output as .csv and Excel sheet

Data Request Web GUI

by Martin Schupfner, DKRZ

- Web GUI using DreqPy to generate customized Data Requests and Volume Estimates for download
- No local installations required
- Planned to use created Data Requests in the Post-Processing Workflow



<https://c6dreq.dkrz.de>

Create Variable List Volume Estimate Documentation & Links Give Feedback Admin Panel

Get Custom CMIP-6 Data Request

Create your customized requested variable List:
This WebGui facilitates the use of Martin Juckes' [Data Request Python \(DreqPy\) API](#) to generate a customized requested variable list for CMIP-6.

Select Endorsed MIP(s)

- All BUT selection
- AerChemMIP
- CdMIP
- CFMIP
- CMIP
- CORDEX
- DAMIP

Select Experiment(s)

- All BUT selection
- All defined by selected MIPs
- All defined by selected MIPs above AND selection
- All defined by selected MIPs above BUT selection
- 1pctCO2
- abrupt-4xCO2
- amip
- esm-piControl
- historical
- piControl

Select maximum priority and tier

Max. ▾ Variable Priority (max)

Workflow



- Averaging
- Interpolation
- ...



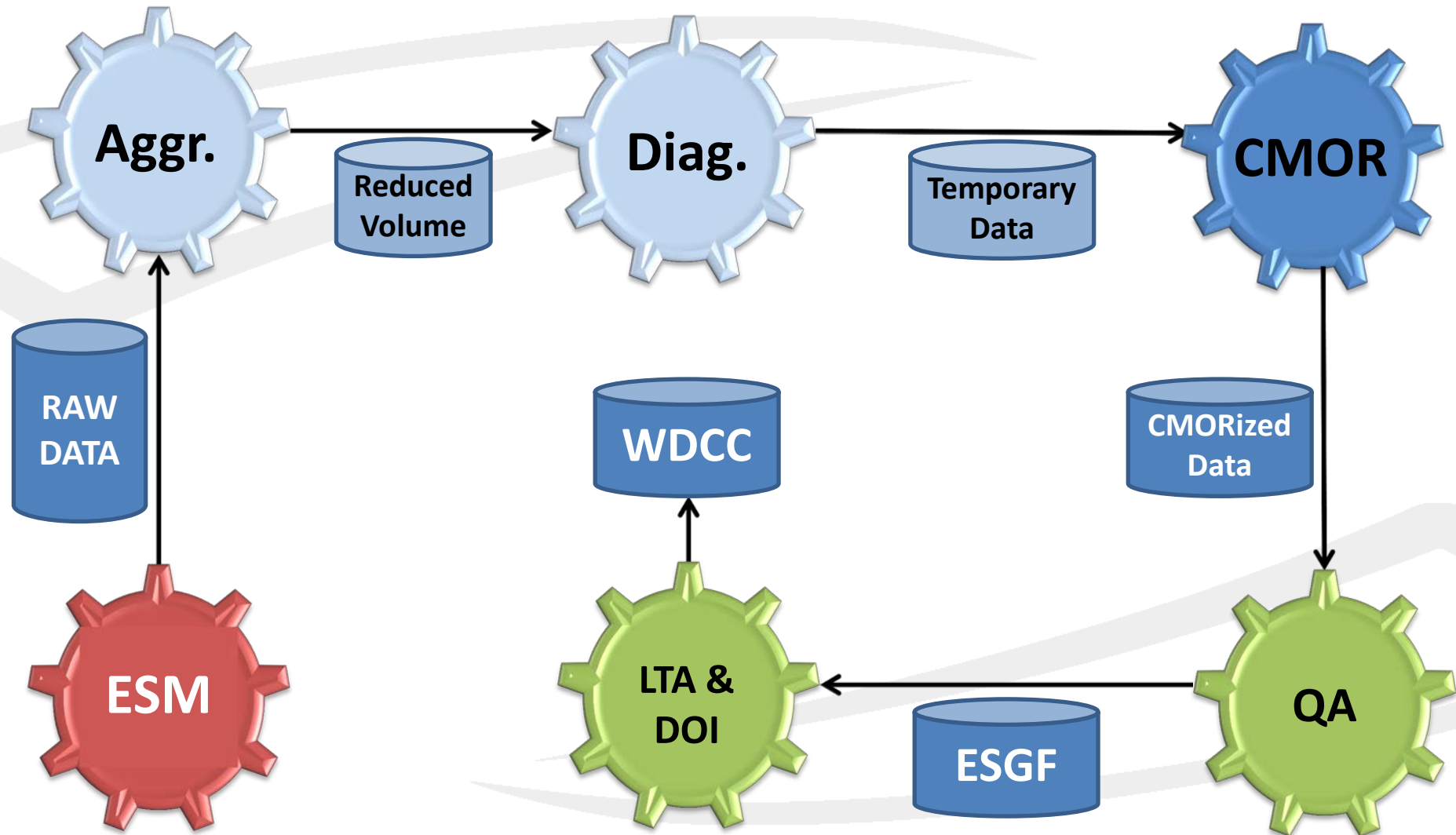
- Diagnostic Scripts
- ...



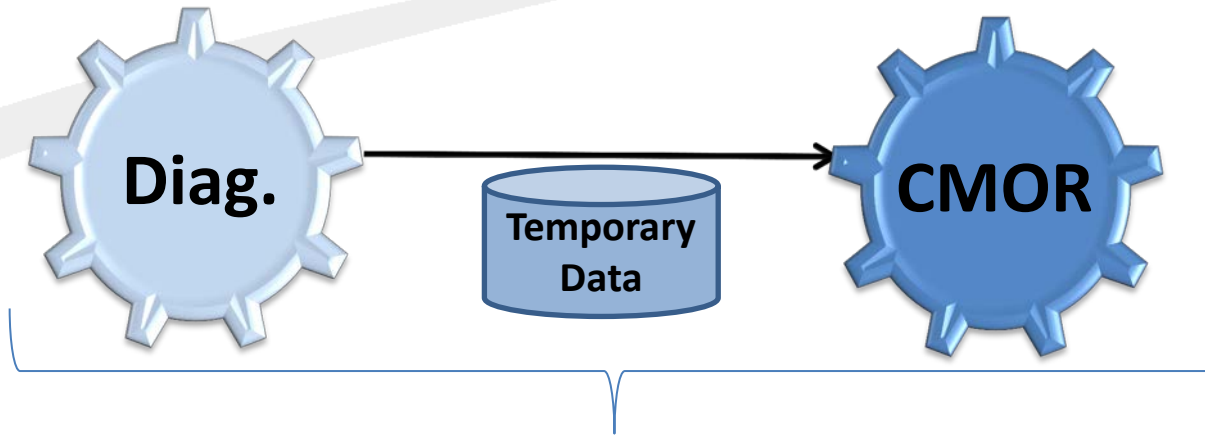
- unit conversions
- Adjust coordinate direction
- Set variable and file attributes
- ...

cdo operator,... (eg. expr / afterburner / ...)

cdo cmor,...

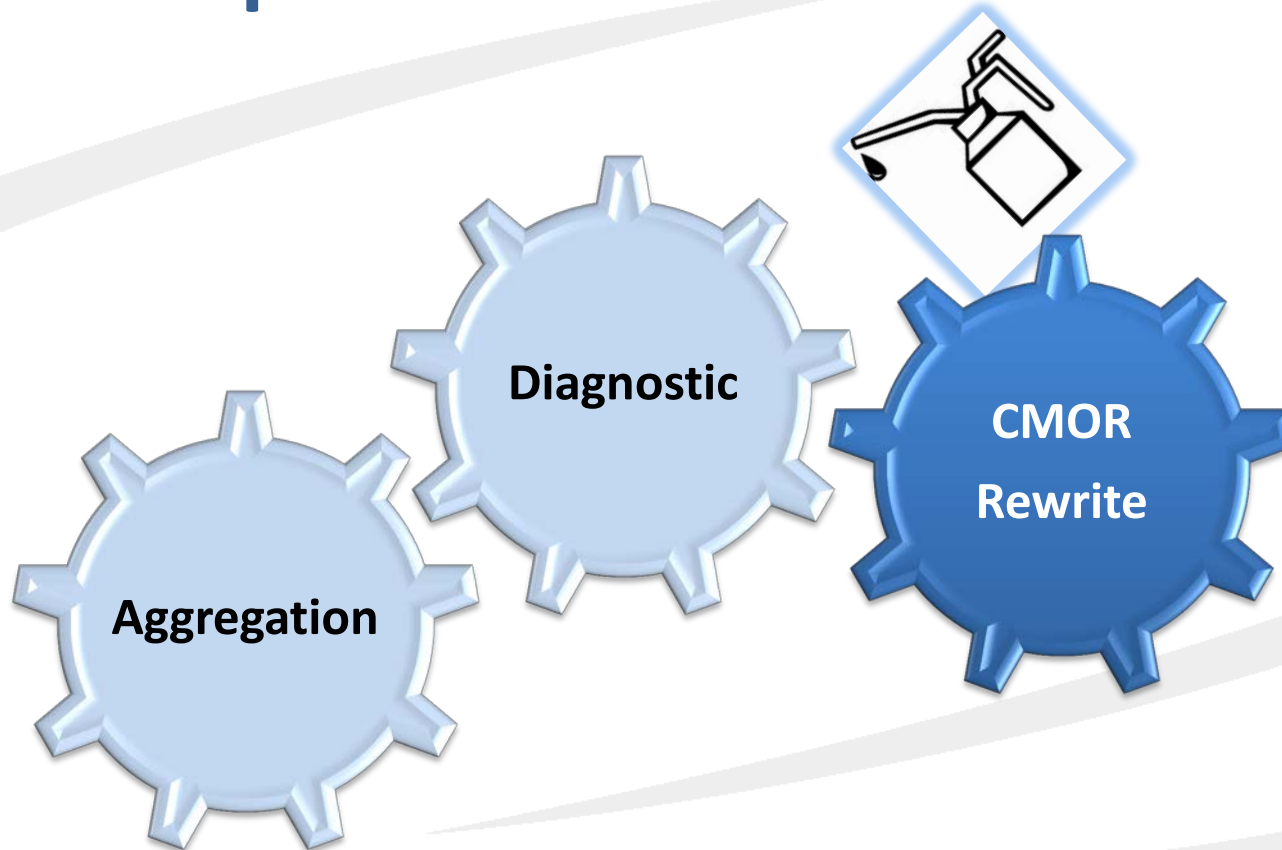


Integration in the Workflow



1. Build template script containing basic Diagnostic and CMORizing for all CMOR Variables
2. Get customized Data Request

Map Model and CMOR Variables



Map Model and CMOR Variables

$\text{Var } X_{\text{model}} \leftrightarrow \text{Var } Y_{\text{CMOR}}$



Operational Diagnostic Recipes

<https://c6dreq.dkrz.de>

Variable Information

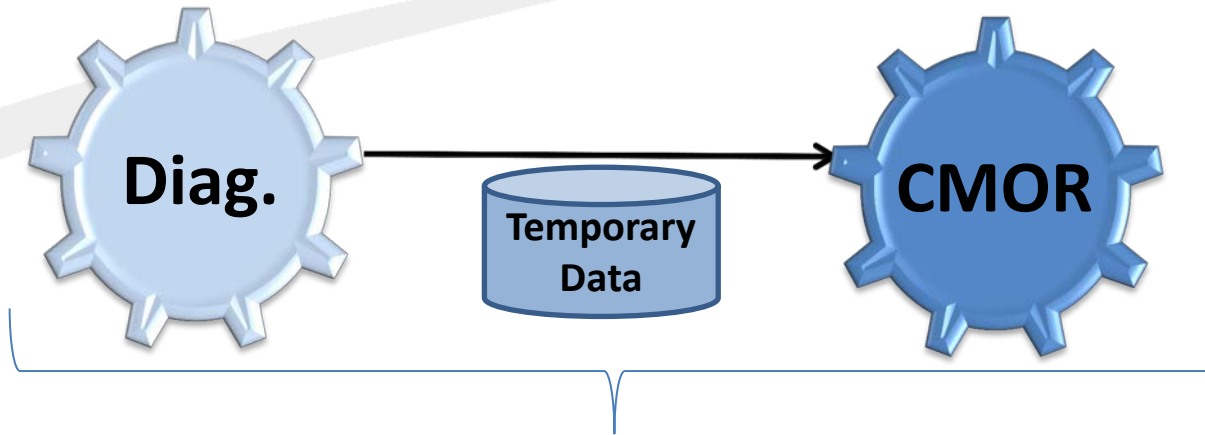
Last Edited	Never
ShortName/CMOR Name	pr
LongName/Title	Precipitation
StandardName	precipitation_flux
Variable Description	includes both liquid and solid phasesat surface; includes both liquid and solid phases. This is the 3-hour mean precipitation flux.
Processing Info	
Realm	atmos
Cell-Methods & Cell-Measures	time: mean (Time Mean) area: areacella
Frequency	3hr
Dimensions & Grid	time longitude latitude
Flux Direction (Positive)	Not applicable
Provenance MIP	CCMI
MIPs (requesting)	DCPP PMIP ScenarioMIP DAMIP CMIP GeoMIP HighResMIP
Experiments (requesting)	G6sulfate DCP-PP-A3 SSP160 DCP-PP-B1 G6solarSlice2 SSP585 ssp245-nat hist-nat past1000 abrupt-4xCO2 DCP-PP-A5 1pctCO2 G1extSlice2 hist-all-aer2 SSP126 G6solar G6slice1 G1extSlice1 G7cirrusSlice2 Coupled G7cirrus DCP-PP-A4 G4-SSA SSP370 SSP585ext SSP245 SSP437 piControl esm-hist ssp245-GHG esm-hist-ext G7cirrusSlice1 hist-all-nat2 SSPXY DCP-PP-A2 historical SSP126ext ssp245-stratO3 DCP-PP-A1 DCP-PP-B22 SSP585extover G6sulfurSlice2 hist-CO2 hist-aer hist-stratO3 esm-piControl DCP-PP-B21 SSP126over Forced-Atmos-Land amip hist-vole hist-sol ssp245-aer Forced-Atmos-Land-2050 historical-ext hist-GHG G1ext
MIP Table	3hr

Variable Mapping Information

- ▶ Examples and Help
- ▶ Old submits

Availability	Model Variable Name	Model Variable Code	Post-Processing Recipe	Variable Units
<input type="checkbox"/> Unavailable <input checked="" type="checkbox"/> Available	_____	_____	_____	_____

Integration in the Workflow



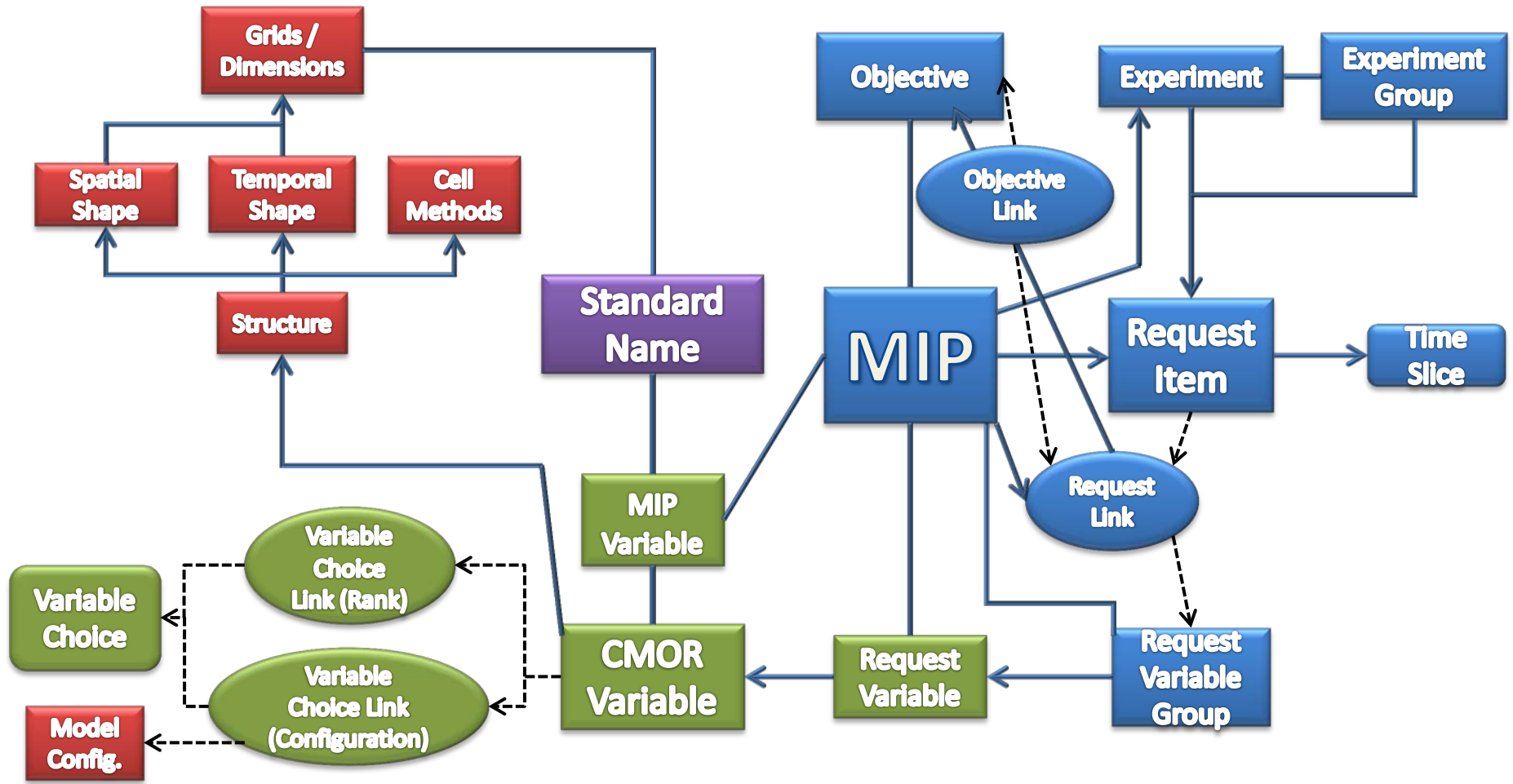
1. Build template script containing basic Diagnostic and CMORizing for all CMOR Variables
2. Get customized Data Request
3. Build Variable Mapping including Diagnostic/Post-Processing Recipes for operational use
4. Filter template script using Data Request and Variable Mapping Information
5. Necessary changes by hand?!



Thanks for your attention!

Questions?

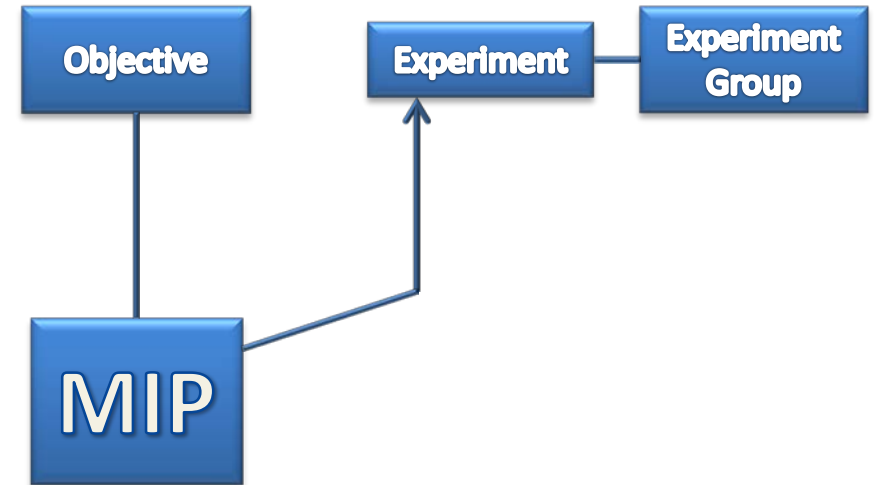
Structure of DreqPy



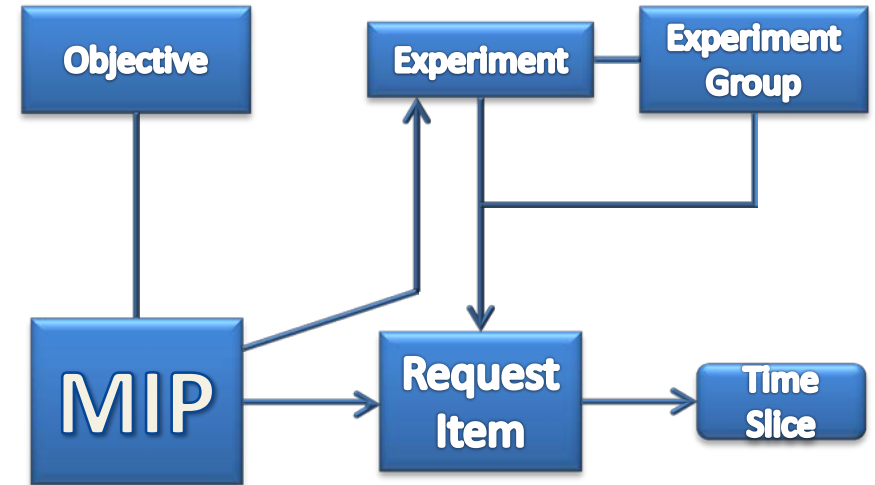
Struktur des Datenrequests



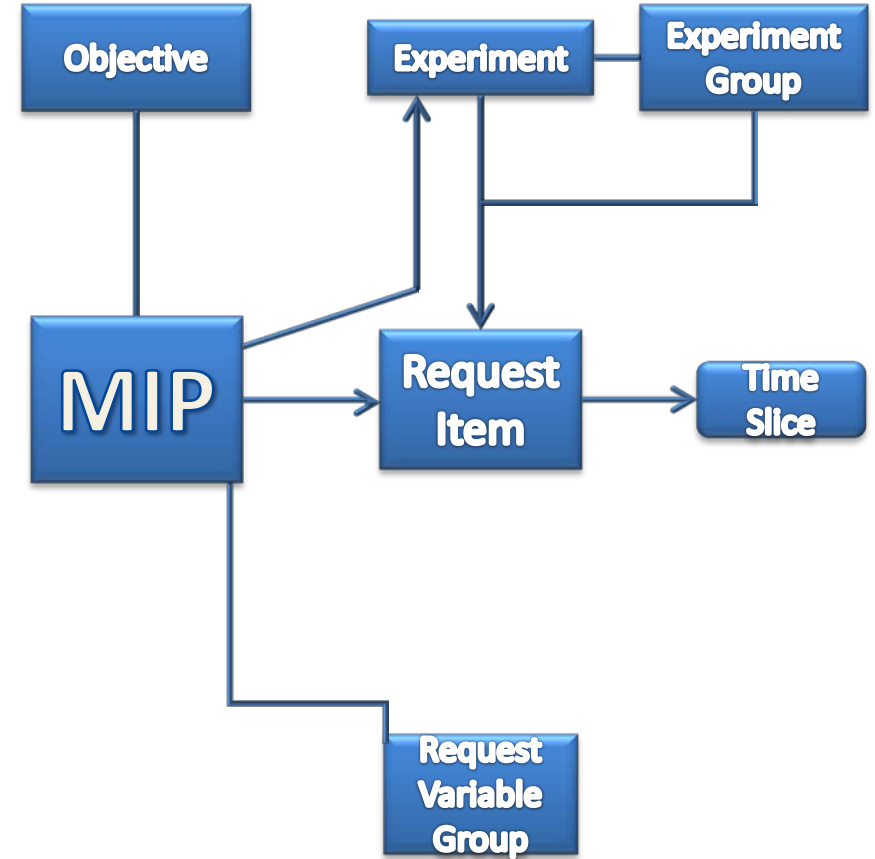
Struktur des Datenrequests



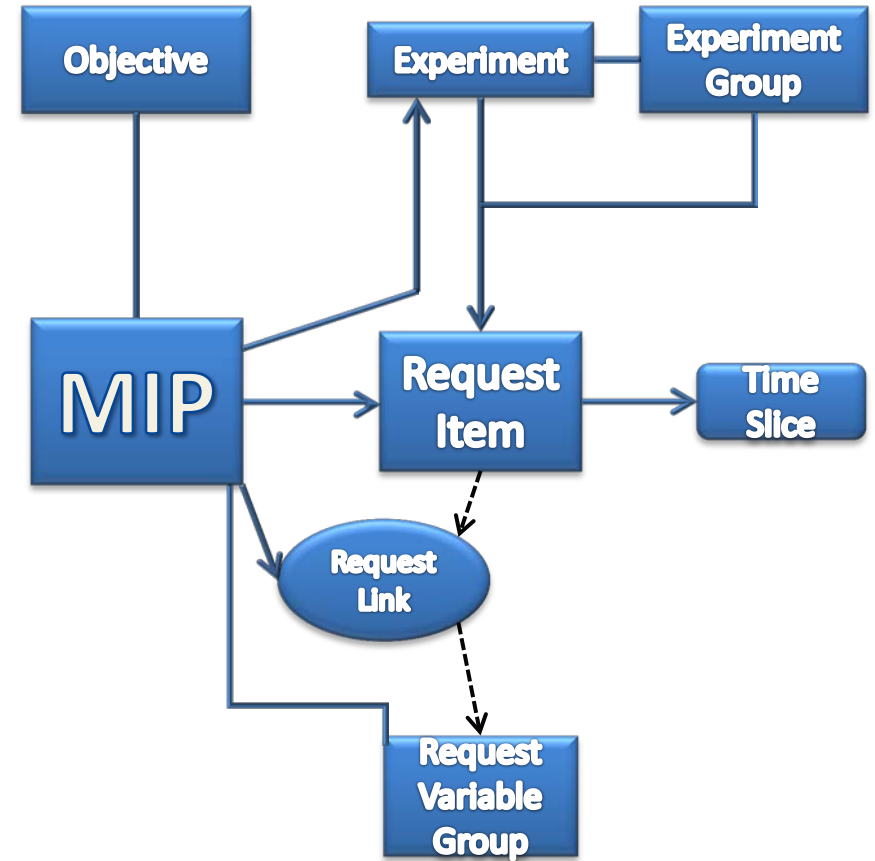
Struktur des Datenrequests



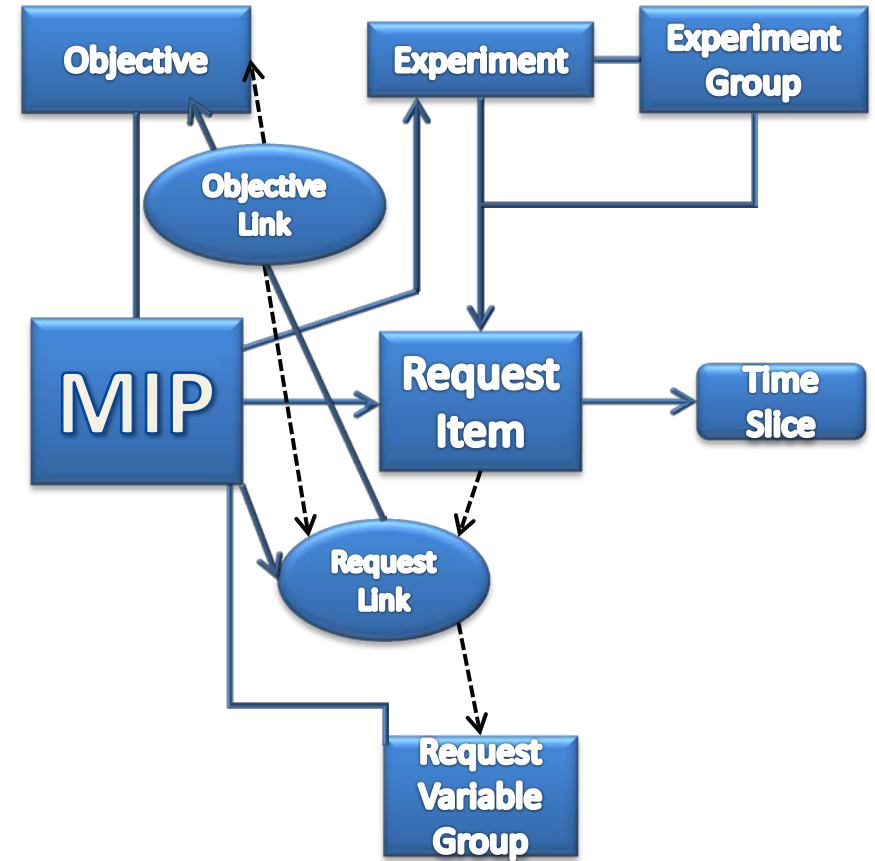
Struktur des Datenrequests



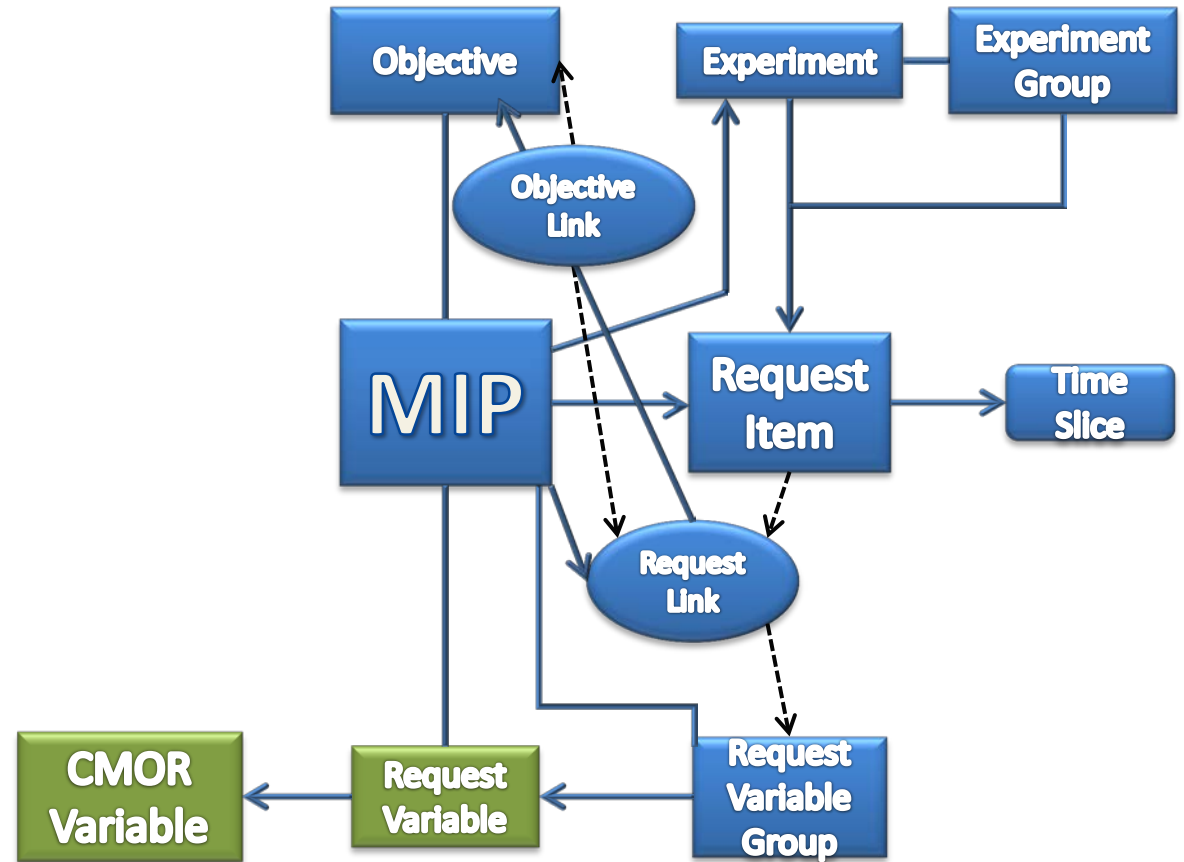
Struktur des Datenrequests



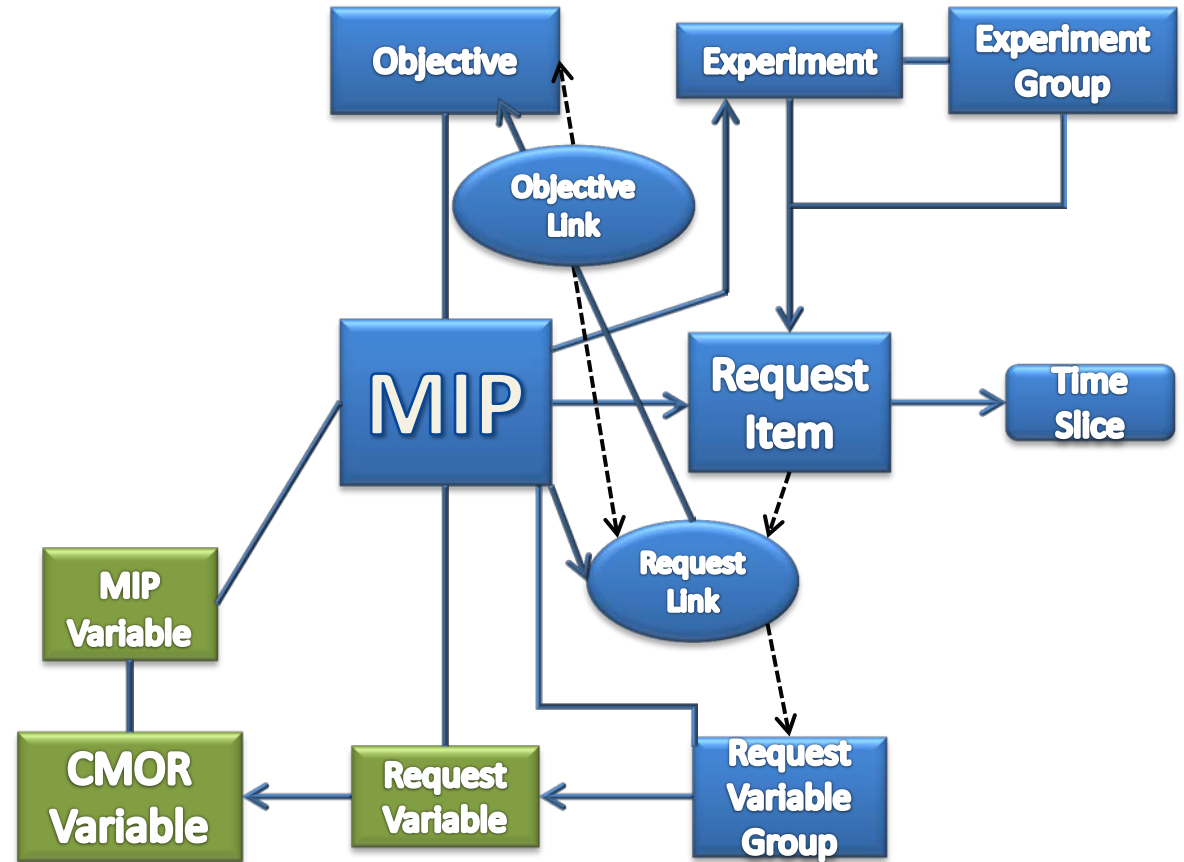
Struktur des Datenrequests



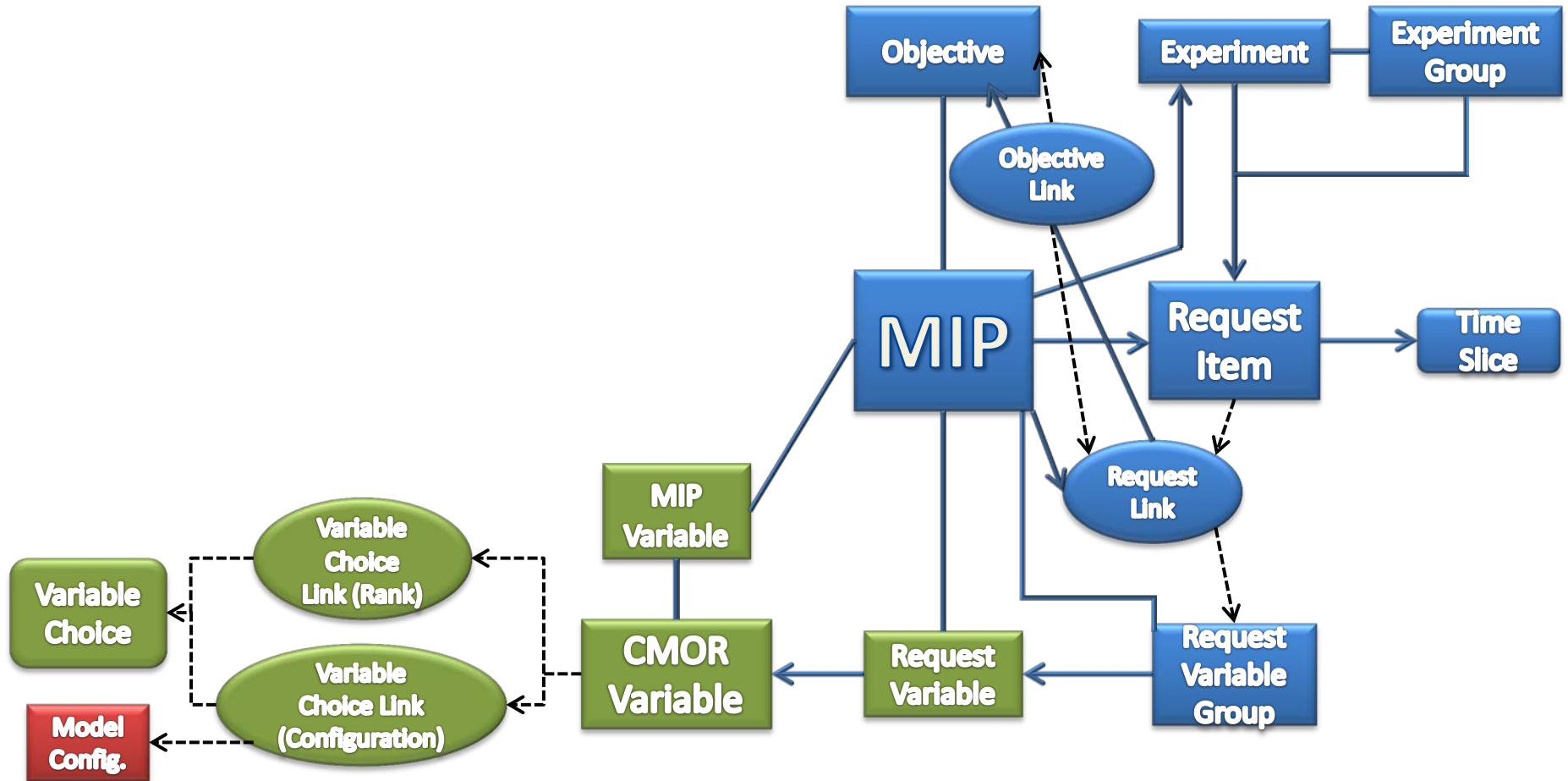
Struktur des Datenrequests



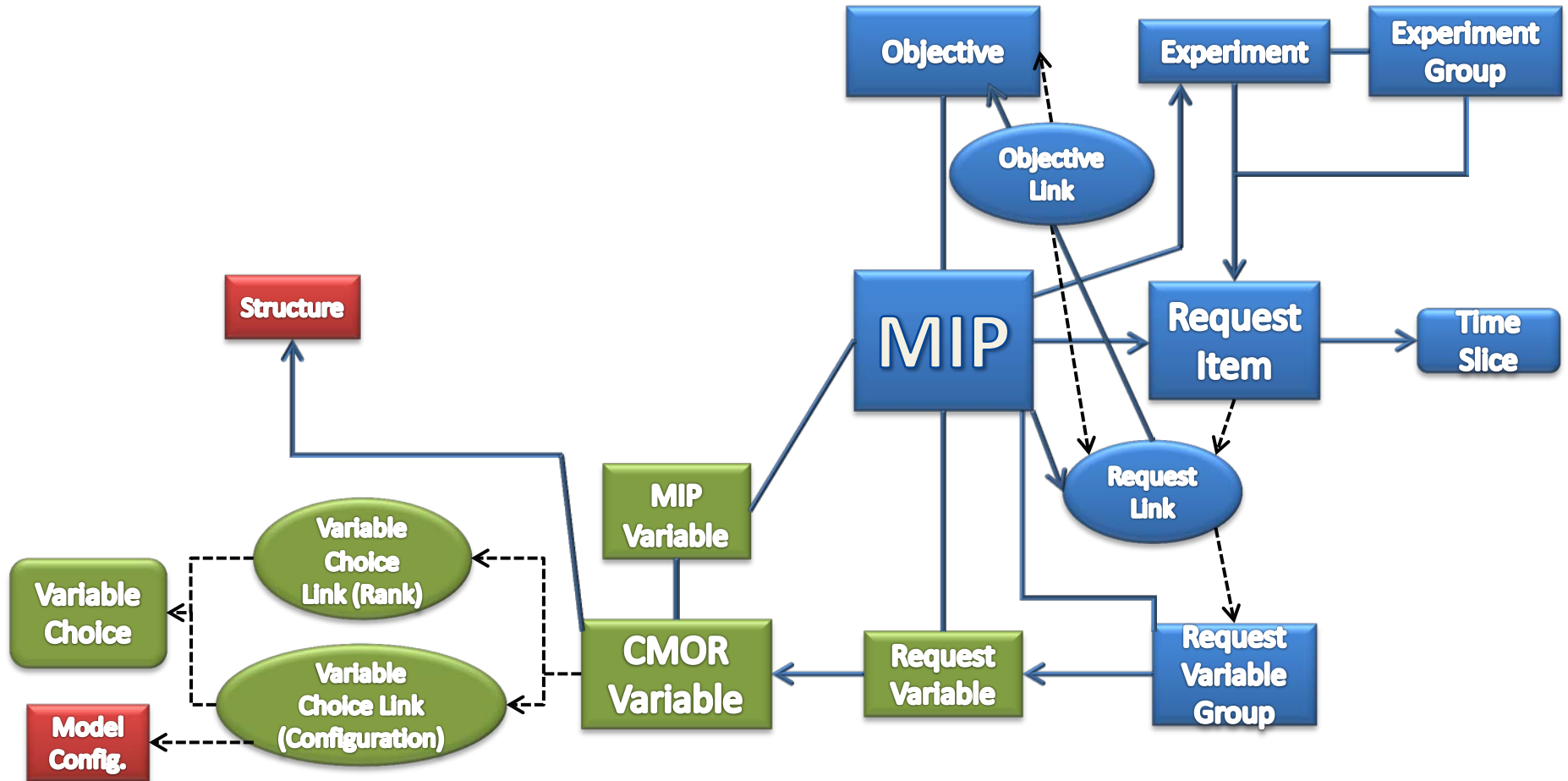
Struktur des Datenrequests



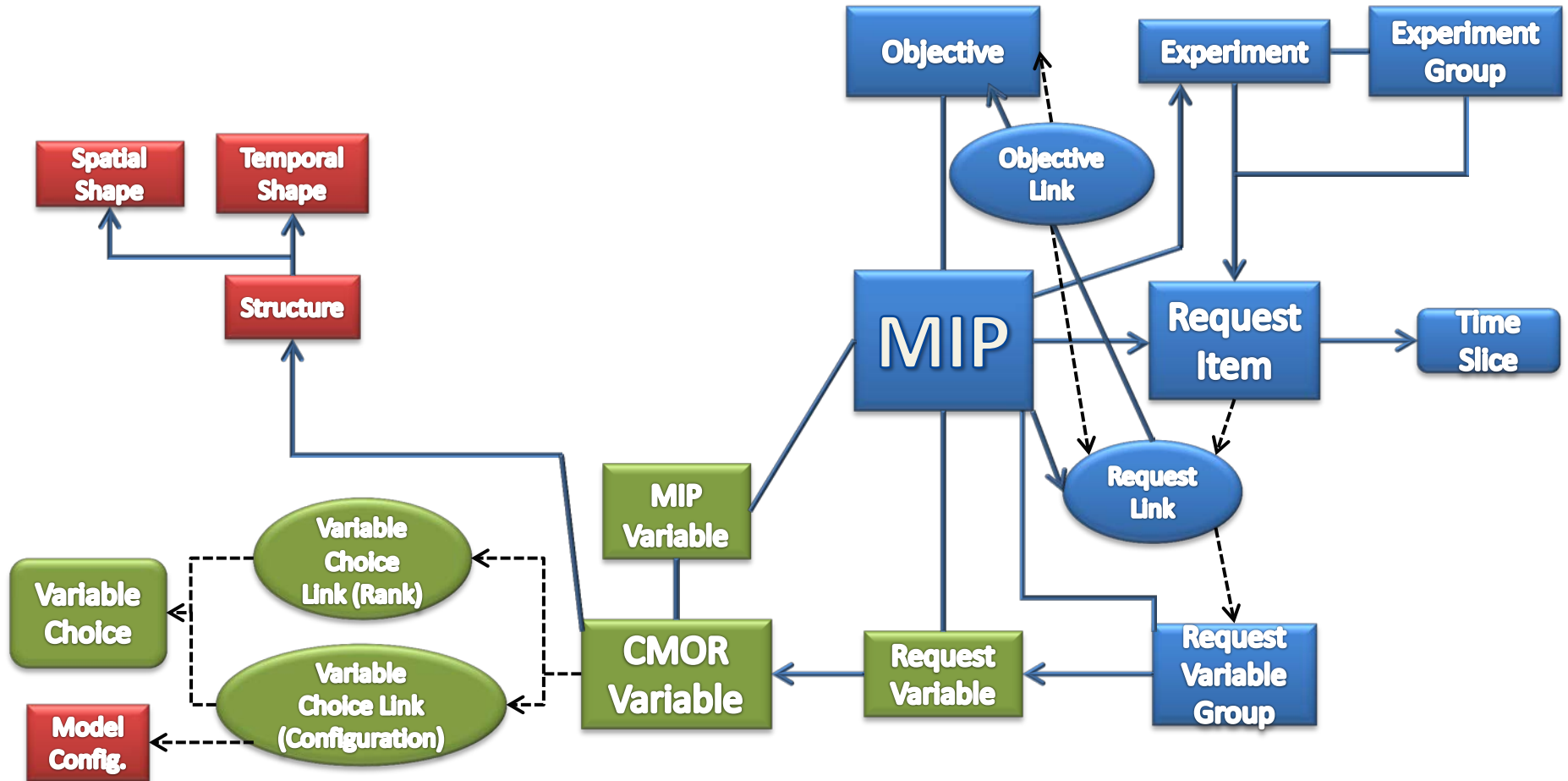
Struktur des Datenrequests



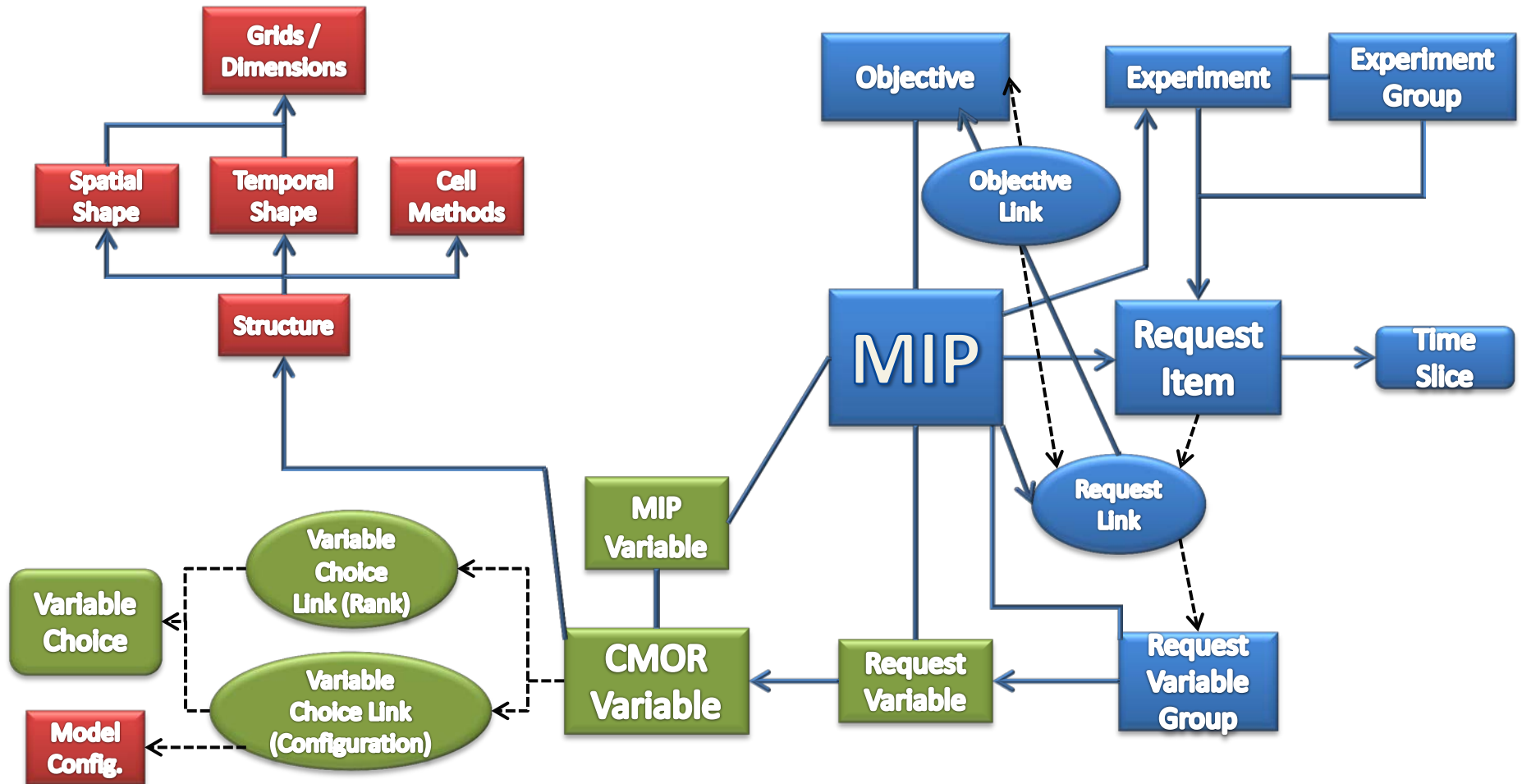
Struktur des Datenrequests



Struktur des Datenrequests



Struktur des Datenrequests



Structure of DreqPy

