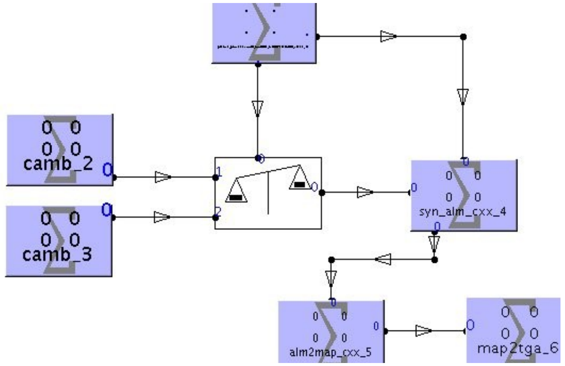
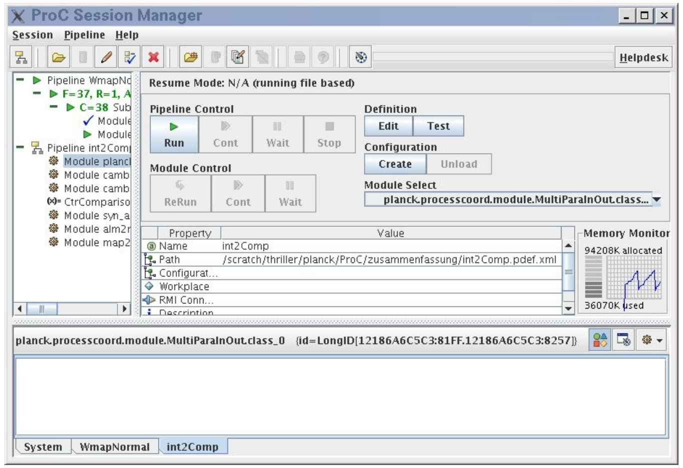


The ProC Scientific Workflow Operation System

The Process Coordinator (ProC) is a versatile scientific workflow engine, which supports graphical construction, execution and control of workflows, consisting of user supplied data processing modules and powerful workflow control structures. It can handle modules written various programming languages within the same workflow, keeping details of their implementation and execution transparent to the scientific user. The ProC is combined with the Data Management Component (DMC), a database application which stores and controls the data products of a workflow. It allows the secure and convenient handling of large scientific data sets, providing fast I/O performance of file systems and the data management capabilities of databases at the same time.

ProC suite (ProC & DMC) was designed for ESA's Planck Surveyor Mission, in order to allow detailed tracking of the production history of each data product – a well-known problem in any complex data processing. Having been kept entirely generic in its application system, it is the ideal infrastructure platform for any scientific project which wants to keep highest standards of integrity and reliability for its scientific data products while aiming at high flexibility and data throughput within a heterogeneous software and hardware environment.

The ProC suite was developed at the Max-Planck Institute for Astrophysics with funding from the Bundesministerium für Wirtschaft und Technologie through the Deutsches Zentrum für Luft- und Raumfahrt. The ProC suite is available for scientific projects and an open source release is foreseen for 2012.



Pipeline WmapNo
F=37, R=1, A
C=38 Sub
Module
Module
Pipeline int2Com
Module planc
Module camb
Module camb
(x)= CtrCompariso
Module syn_a
Module alm2r
Module map2

Resume Mode: N/A (running file based)

planck.processcoord.module.MultiParaInOut.class...

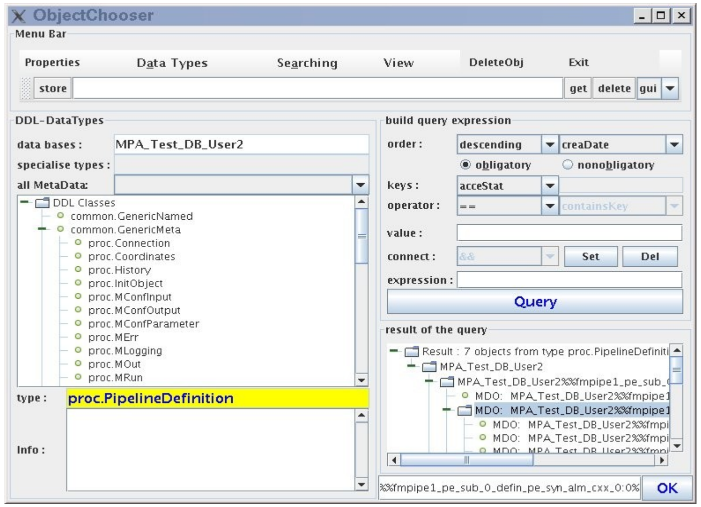
Property	Value
Name	int2Comp
Path	/scratch/thriller/planck/ProC/zusammenfassung/int2Comp.pdef.xml
Configurat...	
Workplace	
RMI Conn...	
Description	

94208K allocated
36070K used

planck.processcoord.module.MultiParaInOut.class_0 {id=LongID[12186A6C5C3:81FF.12186A6C5C3:8257]}

Capabilities of the ProC Workflow Engine

- Graphical design and configuration of workflows, with automatic check of syntactical correctness and completeness
- Control elements for standard processing flows, like loops and conditions, as well as for parallel execution of mutually independent parts of a workflow
- Control element to support sampling problems in high dimensional parameter spaces, like Monte Carlo, minimization or integration
- Resume functionality, which allows the efficient reuse of results obtained in previous runs of a module for the same configuration parameters
- A standardized logging of workflow execution, including the unique identification of all used executables by md5 checksums
- Convenient and flexible handling of resources, like parallel processing capacities, scheduling queues, and remote execution hosts
- Platform-independent implementation in Java and XML
- Computational Grid-enabled through the generic Grid Application Toolkit (GAT)
- Inclusion of programs without code modification possible
- Administrator mode for multi-user ProC installations



data bases : MPA_Test_DB_User2
specialise types :
all MetaData:

DDL Classes
common.GenericNamed
common.GenericMeta
proc.Connection
proc.Coordinates
proc.History
proc.InitObject
proc.MConfInput
proc.MConfOutput
proc.MConfParameter
proc.MErr
proc.MLogging
proc.MOut
proc.MRun

type : proc.PipelineDefinition
Info :

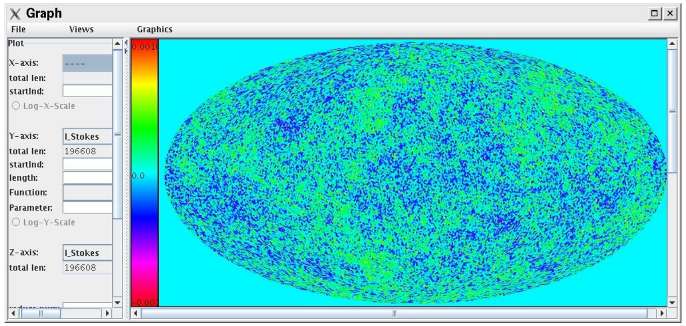
order : descending creaDate
◉ obligatory ○ nonobligatory
keys : acceStat
operator : == containsKey
value :
connect : &&
expression :

Result : 7 objects from type proc.PipelineDefiniti
MPA_Test_DB_User2
MPA_Test_DB_User2%%fmpipe1_pe_sub_
MDO: MPA_Test_DB_User2%%fmpipe1
MDO: MPA_Test_DB_User2%%fmpipe1
MDO: MPA_Test_DB_User2%%fmpi
MDO: MPA_Test_DB_User2%%fmpi
MDO: MPA_Test_DB_User2%%fmpi

%%fmpipe1_pe_sub_0_defin_pe_syn_alm_cxx_0:0%

Capabilities of the Data Management Component (DMC)

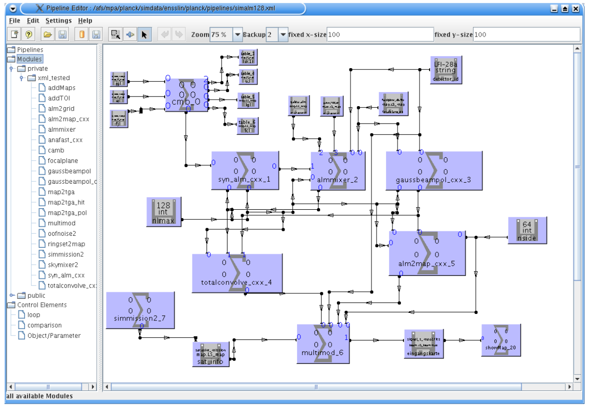
- Ability to connect to different database management systems (Oracle, PostgreSQL, MySQL, …)
- Generic interface for different programming languages (C, C++, Fortran, Java, IDL)
- Automatic tracking of history information for all data objects (i.e. information about the data, parameters, and executables which were used to create them)
- Graphical tool to browse data objects, providing easy access to data and metadata
- Versatile data model permits user defined data type definitions with object oriented inheritances
- GUI based query construction and reusage



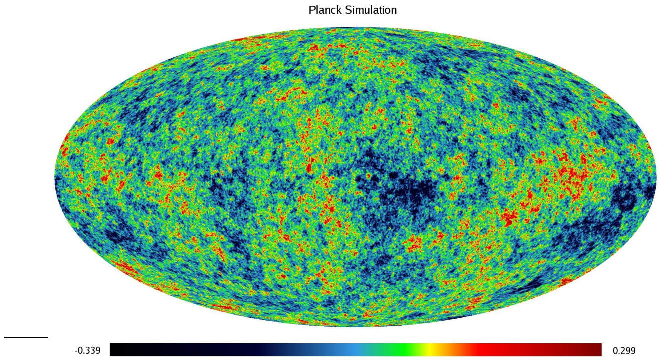
X-axis: ----
total len:
startInd:
Y-axis: I_Stokes
total len: 196608
startInd:
length:
Function:
Parameter:
Z-axis: I_Stokes
total len: 196608

Simulation and Data Analysis

Within the Planck project, the ProC suite is extensively used for data analysis and scientific simulations. The image shows a typical Planck simulation workflow in its ProC editor view. Programs, data flows through the DMC, as well as data inputs and outputs are visible.

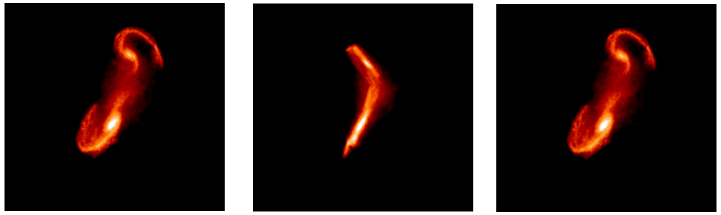


The editor view permits easy modification of the many parameters configuring the individual programs. ProC can selectively recalculate those parts of the workflow, which are affected by such a change. The workflow can be extended by inclusion of further programs as displayed on the left hand panel. One of the outputs of this workflow is a simulated CMB sky, as shown below.



Scientific Sampling

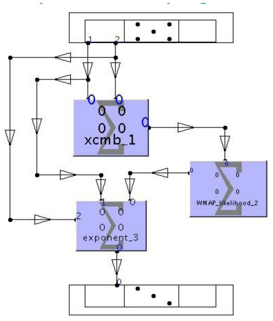
Scientific inference often involves the statistical comparison of the result from a complicated workflow calculation with some real data. Many of the parameters configuring the workflow are to be determined by such costly computations. A large number of algorithms are in principle available to search through or to integrate over the high-dimensional parameter spaces. However, since adapting them to a given problem is expensive, very few are usually applied in practice. The ProC has built-in sampling control functionality, which permits the user to apply a large number of predefined or user-supplied sampling algorithms to any parameter space of any ProC workflow. The individual sampling algorithms can easily be interchanged, permitting investigations of the optimal sampling strategy for a given problem. Template algorithms in Java exist, which can be adapted by scientific programmers, as well as a multi-lingual API for the inclusion of C, C++ and Fortran sampling codes. The library of predefined algorithms embraces simple grid search, sparse grid optimisation, simplex search, Metropolis-Hastings Markov Chains, nested sampling, and is continuously extended.

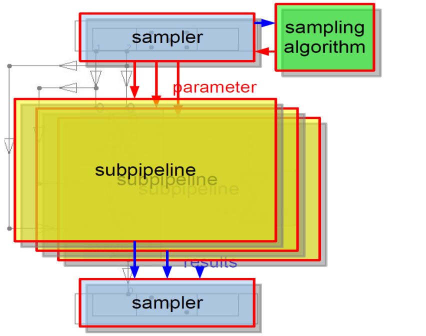


An astronomical example of such a scientific inference problem is the reconstruction of the initial conditions of interacting galaxies. The left panel shows a virtual observation of a simulated galaxy. The middle panel shows another simulated galaxy with randomised initial conditions. And the right one a simulated galaxy after ProC-optimisation of the initial conditions in order to reproduce the initial target image (left). A simple simulated annealing algorithm was used.

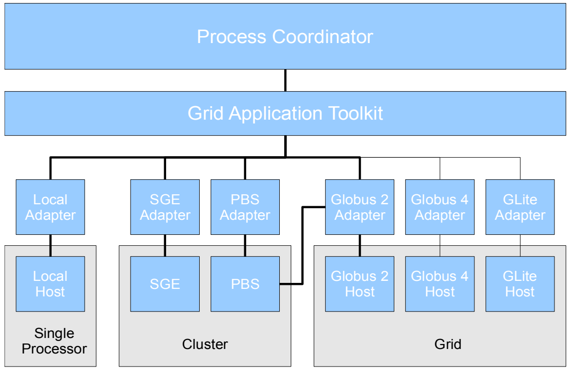
Parallel Execution on the Grid

Many sampling problems are embarrassingly or partly parallel. The ProC helps to exploit parallel computing power ranging from multi core CPUs, over computing clusters via scheduler, up to computational grids for parameter sampling problems, as well as for all other kind of workflows. A sampling workflow, as shown below, which is controlled by an algorithm providing swarms of parameters, will be executed in parallel.





The execution host is reached through the Grid Application Toolkit, which contains adaptors for a large variety of execution environments. The same workflow, initially developed on a laptop, can therefore easily be transferred onto powerful computational platforms.



Contact: Torsten Enßlin <ensslin@mpa-garching.mpg.de>
Max Planck Institute for Astrophysics
Karl-Schwarzschild-Str. 1
85741 Garching, Germany
phone: +49 89 30000 2243
fax: +49 89 30000 2235

Gefördert durch:

Bundesministerium
für Wirtschaft
und Technologie

aufgrund eines Beschlusses
des Deutschen Bundestages