

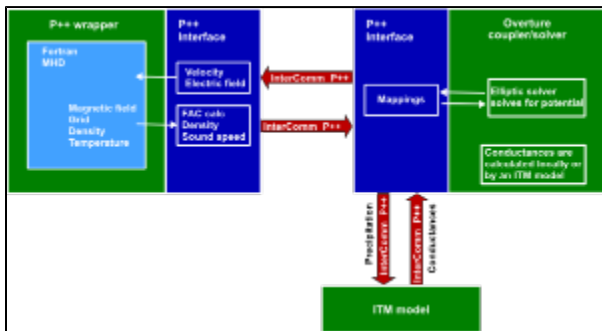
# MIX

## MIX: Magnetosphere Ionosphere Coupler/Solver

The MIX code's primary function is to provide the inner boundary condition for global MHD models, but it serves in fact a much more diverse purpose. It can utilize existing empirical models of ionospheric conductance to obtain the electrostatic potential solution (solver mode), or it can receive conductances from an external ionosphere-thermosphere model and then compute the potential (solver/coupler mode). In the future, it is planned that this coupler will mediate communication between other geospace component models as well, i.e. Rice Convection Model (RCM). The major design consideration in developing this code was to adopt the loose coupling concept, i.e. plug-and-play should be made as simple as possible. Two software packages are utilized in order to achieve this goal:

- [InterComm-1.6](#), developed by Alan Sussman's group at the University of Maryland at College Park. InterComm provides means for data communication (no file dumps!) between different executables which can be parallelized and run on different systems.
- [Overture](#), an object-oriented framework for solving partial differential equations on overset grids developed by Bill Henshaw at LLNL. Both packages make extensive use of [A++](#), a set of C++ classes allowing vector array operations in a serial/parallel environment.

The MIX code is not so much an Overture code as it is an A++ code. Overture's functionality is used in two places only: to interpolate between grids and to obtain the elliptic solve for the ionospheric potential. The caveat is that Overture's overset grid technology, something that the software was designed for, is not very useful when one needs to interpolate between entire grids (the problem that we are dealing with). The problem that Overture is solving is, in a way, much more sophisticated - to compute interpolation boundaries. Fortunately, Overture does have simpler interpolation functions, i.e. to interpolate from a grid onto a set of unstructured points, and this is exactly what we are using. Any interpolation software could be used for this purpose and we could in principle use our own interpolation functions too. The second place where we use Overture is where its use is more justified. The application of Overture to obtain the ionospheric elliptic solution makes this code very flexible and potentially gives nice opportunities for interesting physical experiments: one can play all kinds of games with different boundary conditions, with the location of the low-latitude boundary, the grid resolution, etc. In addition, Overture can interact with a number of sparse matrix solvers (Yale, Harwell, SLAP, and PETSc) and a variety of matrix inversion algorithms in combination with different preconditioners can be used. An option unexplored so far, but a very attractive one, is the use of multigrid solvers which are also implemented in Overture.



The picture above demonstrates the general structure of MIX and its communication with other components. The box on the left represents an unspecified global MHD model. The P++ interface in the MHD code is not really a part of the MIX code (maybe it should be such) but it is what actually talks to the MHD model. This interface calculates the field-aligned current, the density and the sound speed of plasma at the MHD inner boundary and sends these via InterComm communication channel to the MIX code. After the ionospheric potential solution is obtained, it is sent back to the MHD interface which calculates plasma transverse velocity and electric field and gives it back to the MHD code. The only requirement to the MHD code so far is that its data be defined on a logically spherical grid, but it is anticipated that an Overture module will be developed to reduce this condition to an arbitrary grid. The interface is a fully parallel P++ code, and thanks to the InterComm beauty, the MIX code does not care about it. The structure of the MIX code (box on the right) is similar. It has an interface (actually, more than one - see the rest of the documentation) which talks to an MHD or an ITM model (and maybe something like RCM too), does all the data massaging, conversion of physical variables, interpolation between different grids, etc. If needed the ionospheric conductances may be calculated locally using an empirical model (the conventional LFM faxed-memo-kinda-AMIE-model or the Moen-Brekke, 1993 can be used). Then all the data required are passed to the elliptic solver which does the solve, hands the solution back to the interface, the data required is extracted and is sent back to the components that need it. Simple as that... Enjoy!

This code development was supported by the Center for Integrated Space Weather Model (CISM) funded by the National Science Foundation STC program under agreement ATM-012950.