Progress Towards An Improved High-Fidelity Forecasting Capability using Combined Mesoscale and Microscale Models

Presented by

William J. Coirier, Ph.D.

at the

2007 Chemical and Biological Information Systems Conference

Hazard and Environmental Modeling Session

Austin, TX

Thursday, January 11, 2007

Outline

- **Summary of the SBIR Phase I Findings**
- **Overview of coupling framework using MCEL**
- **CFD Model and Related Software Development Overview and Progress**
    - **Cartesian Adaptive Mesh Refinement-based Virtual Cell Embedding**
    - **Specialized Parallel RANS Solver**
    - **Wind library database and common file format**
- **Work plan**

Acknowledgements

- This work is funded under a Small Business Innovation Research Phase II grant:
    - CDR Stephanie Hamilton/USN
- NCAR:
    - Dr. Fei Chen, Dr. John Michelakes, Dr. Jimi Duddhia
- Bettencourt Consulting (MCEL):
    - Dr. Matt Bettencourt
- CFDRC:
    - Sura Kim, Shawn Ericson, Saikrishan Marella, Joel Mayes

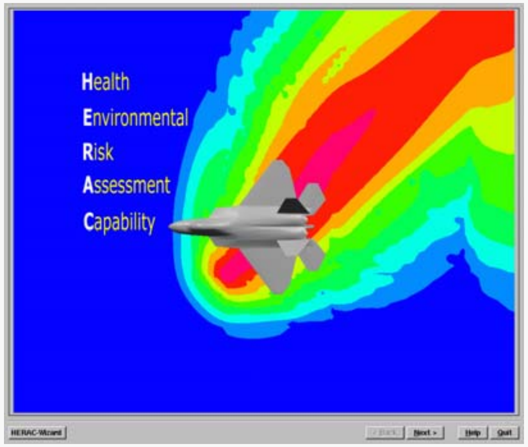
Defense Applications Branch

Mission

Support the DOD, DHS, DoE and Industrial Customers via Technology Transfer of First-Principles Based Scientific Computing Applications and Methodologies

AFFTC Edwards AFB

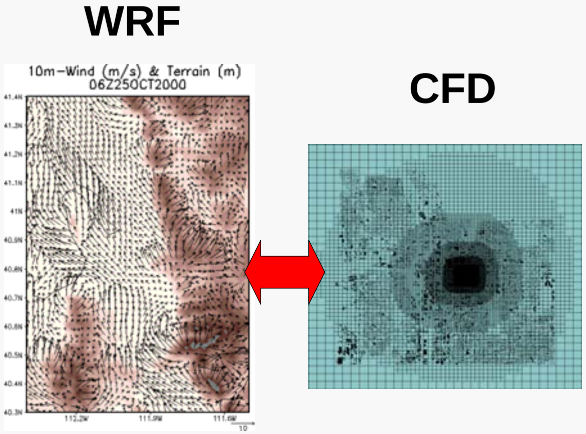
Health and Environmental Risk Assessment Capability: HERAC



CFD-based Risk Assessment for A/C Maintenance

DTRA

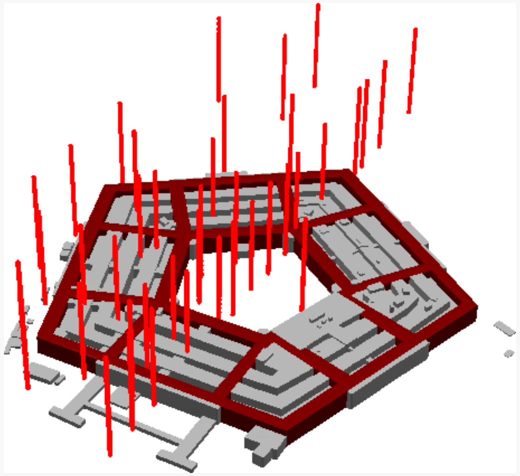
Coupled Micro- and Mesoscale Weather Models



Improved High-Fidelity Microscale T&D via Coupled NWP and CFD

PFPA/DARPA

Pentagon Shield CFD Model Component



Operational Force Protection System to Guard against CBRN/E

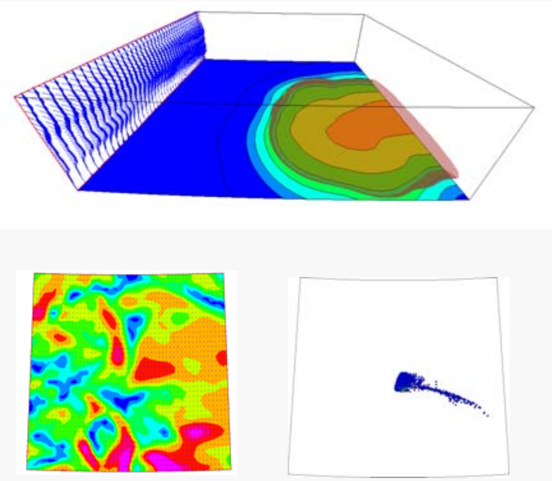
Defense Applications Branch

Mission

Support the DOD, DHS, DoE and Industrial Customers via Technology Transfer of First-Principles Based Scientific Computing Applications and Methodologies

DTRA

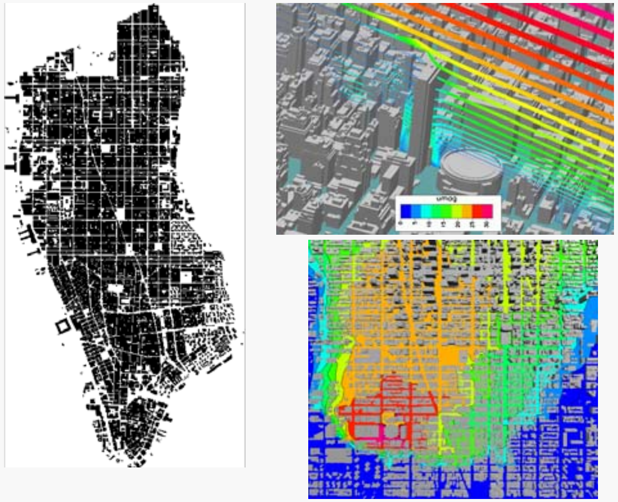
Real-Time 3D Visualization Capability



Client/Server Visualization Capability for Next Generation Consequence Assessment Models

DHS/DTRA

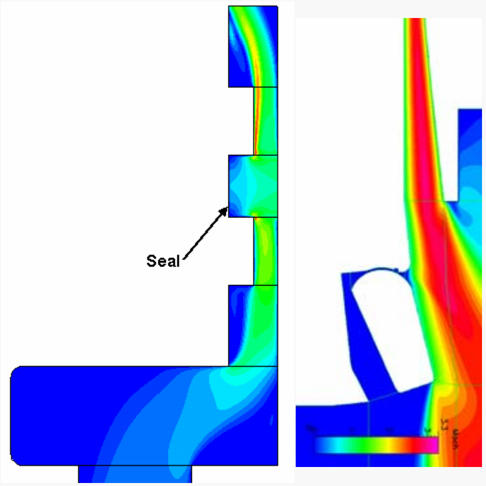
CFD Support for National Field Tests



Transport and Dispersion Studies Pre- and Post-Test, Urban Dispersion Program (UDP)

NASA

CEV Heat Shield Gap Analyses



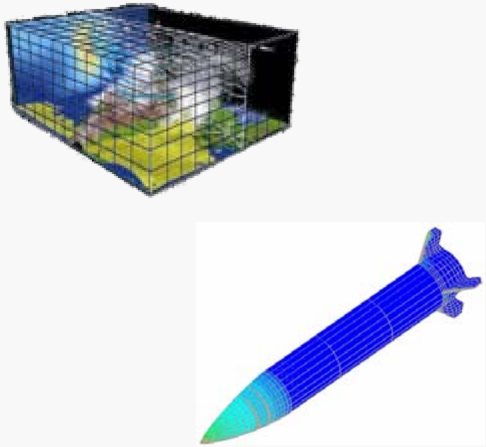
Ablator Gap Seal Flow and Heat Transfer Analyses

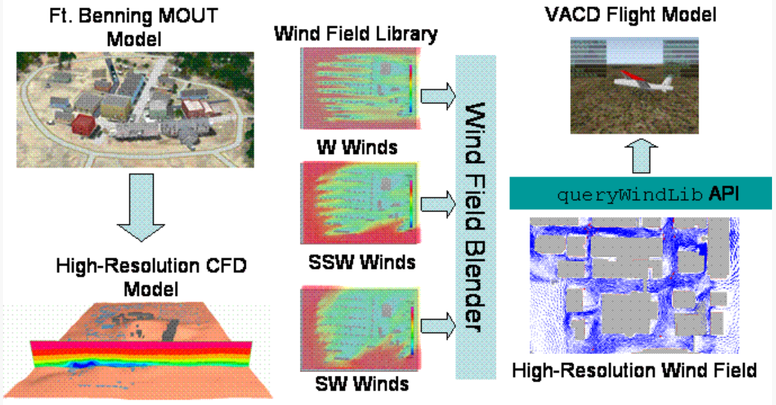
Defense Applications Branch

Mission

Support the DOD, DHS, DoE and Industrial Customers via Technology Transfer of First-Principles Based Scientific Computing Applications and Methodologies

Army	AFRL/VACD	Private Firm
Missile Weather Encounter Modeling Software	**High Resolution Micro-UAV Wind Fields**	**Building Environmental Impact Study**





Couple GCAT output and ATAC Missile Model: Coatings, Ablation, Hydrometeor Impact…

CFD Wind Libraries for use by Micro-UAV Flight Vehicle Models

Determine effect of potential building in Manhattan

WRF + CFD-Urban: Investigate Improved T&D Capability

SBIR Phase I: "*Improved High-Fidelity Forecasting Capability using Combined Mesoscale and Microscale Models*"

  - **Tech. Monitor: CDR Stephanie Hamilton**

- **Investigate improvement in T&D accuracy via merging capability of Mesoscale and Microscale Models**
- **Focus upon community models:**
  - **Weather Research and Forecasting Model: WRF**
- **Evaluate Downscale Data Transfer upon CFD-Urban T&D Accuracy:**
  - **Use Urban 2000, IOP 10, Compute Statistical Measures**
    - **Raging Waters Met Station input (baseline)**
    - **WRF Forecast Mode: Noah and Noah/UCM Urban Parameterization Schemes**
- **Investigate Upscale Data Transfer: Compare WRF and CFD-Urban Fields**
- **Demonstrate Operational Concept:**
  - **Cyclical Met Data Ingest using Event-Driven CFD-Urban**

CFD-Urban: Urban Area T&D Model

Computational Fluid Dynamics Modeling for Wind, Turbulence, Transport and Dispersion in Urban Areas

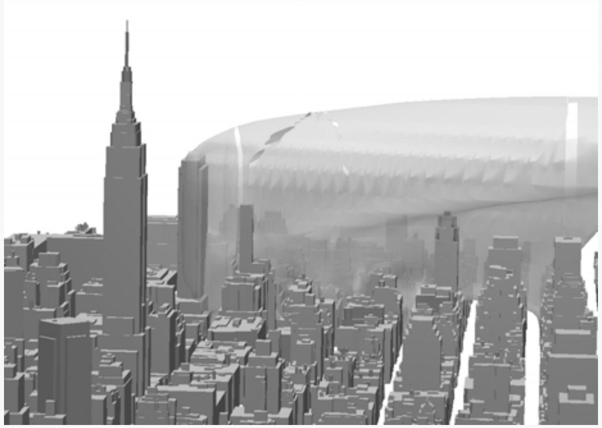
Specialized Model Generation, Setup, Processing

Building Models: GIS, Lidar/imagery, CAD

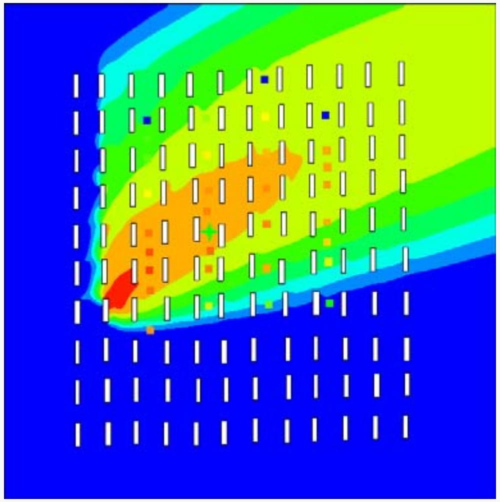
Flow and Turbulence: Steady/Unsteady, RANS, LES

Transport: Eulerian (gases), Lagrangian (particles)

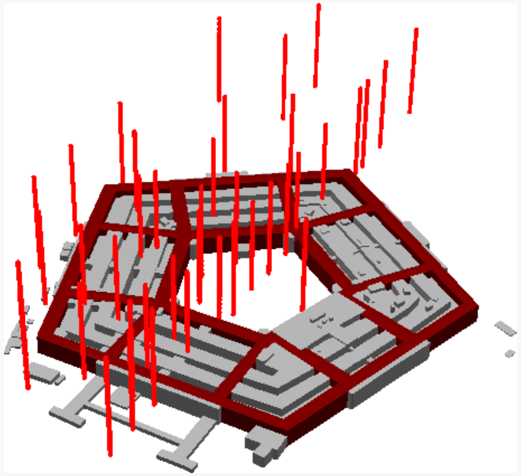
Vertical Mixing, Lateral Spreading, Turbulence Generation



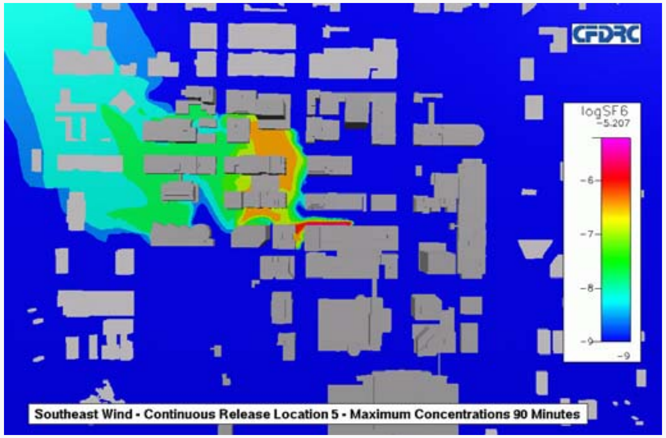
MSG05 and MID05 Support



MUST



Pentagon Shield Model



JU2003 Support

T&D Model Accuracy Characterization: IOP 10 Urban 2000

- **Urban 2000: Field Test conducted in Salt Lake City**
  - **SF6 released in Central Business District**
  - **Samplers located in CBD and on “arcs” located downstream**
  - **WRF Forecast Corresponding to IOP 10**

•Statistical Comparison of Predicted to Measured Concentration Data on arcs noted

$$\mathrm{FB}=\frac{\left(\overline{C_o}-\overline{C_p}\right)}{0.5\left(\overline{C_o}+\overline{C_p}\right)}$$

$$\mathrm{MG}=\exp\left(\overline{\ln C_o}-\overline{\ln C_p}\right)$$

$$\mathrm{NMSE}=\frac{\overline{\left(C_o-C_p\right)^2}}{\overline{C_o}\;\overline{C_p}}$$

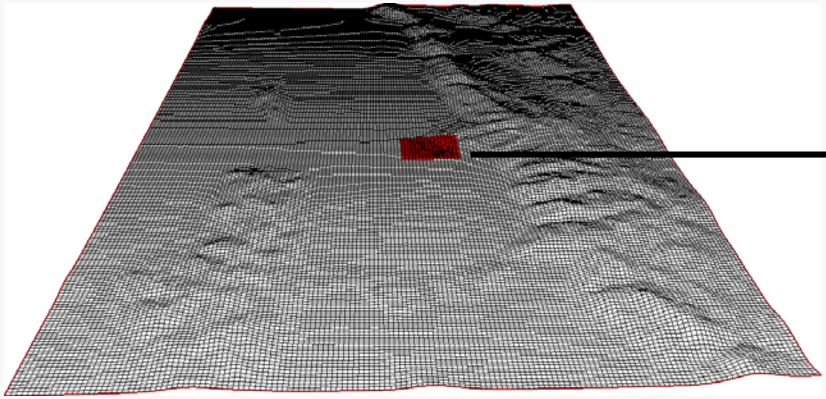
$$\mathrm{FAC2}=\text{fraction of data that satisfy }\; 0.5\le\frac{C_p}{C_o}\le 2.0$$

$$\mathrm{VG}=\exp\left[\overline{\left(\ln C_o-\ln C_p\right)^2}\right]$$

Acceptable values:

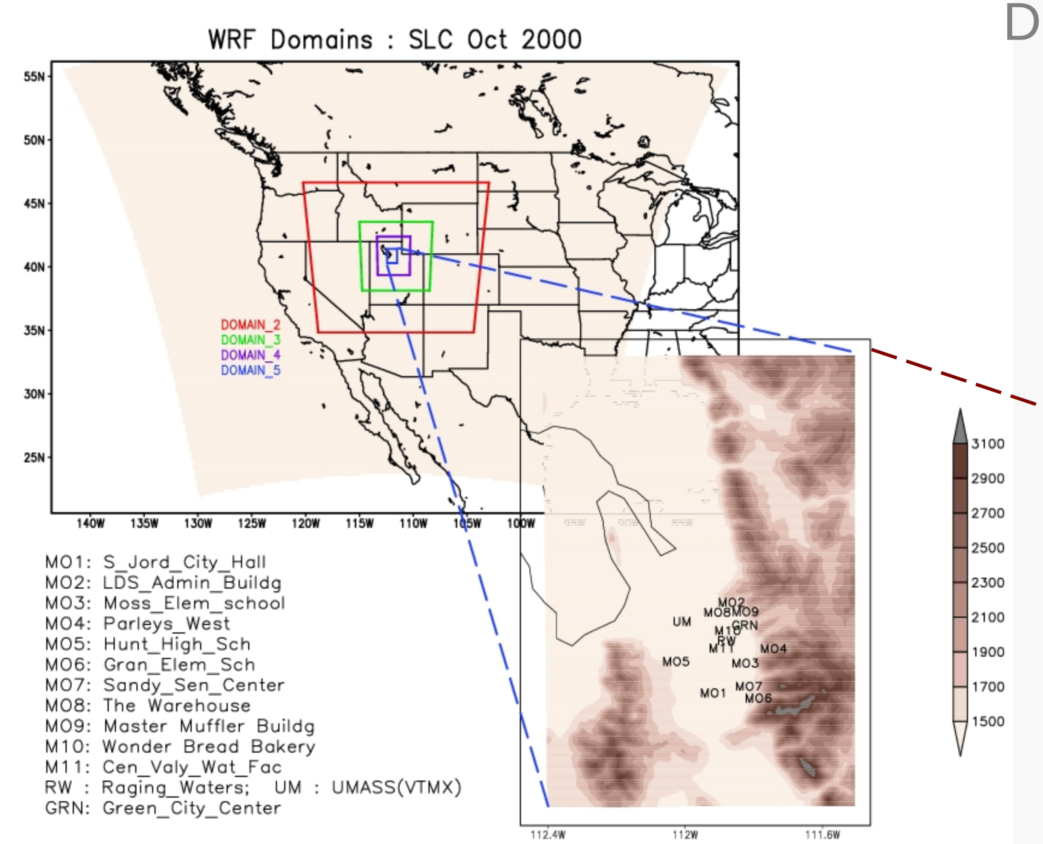
- **FAC2 > 0.5**
- **-0.3 < FB <0.3 (0.7 <MG < 1.3)**
- **NMSE < 4 (VG <1.6)**

WRF Lowest Grid Level



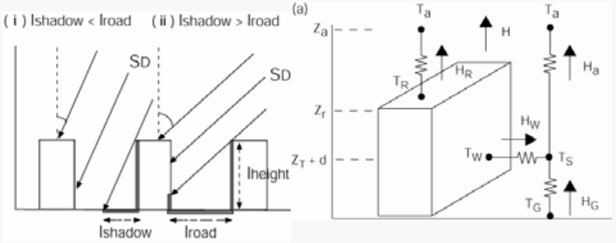
WRF Forecast Model Runs: Urban 2000 IOP 10 Period

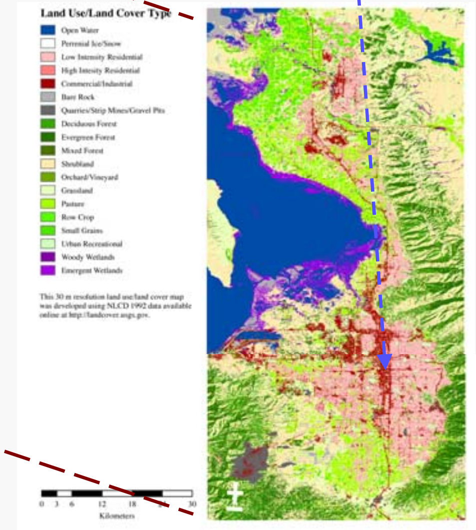
Domains: 40.5,13.5,4.5,1.5,0.5 Km



Complex terrain on WRF nested D-5 (0.5 km grid spacing) over the Salt Lake City area

Single layer Urban Canopy Model





Complex urban land use distribution over SLC

WRF to CFD-Urban: Downscale Transfer Procedures

- **Process WRF datasets (NetCDF format):**
  - **Interpolate data to CFD-Urban grid boundary faces**
  - **Continuous, linear interpolant**

$$f_L = \sum_{n=1}^{4} N_n f_n \qquad f_U = \sum_{n=5}^{8} N_n f_n \qquad f(x,y,z) = f_L + (f_U - f_L)\frac{(z - z_L)}{(z_U - z_L)}$$

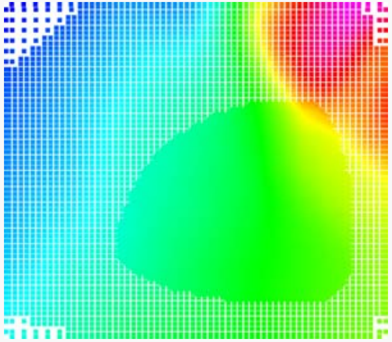
- **Pressure: Remove hydrostatic variation by subtracting ideal atmosphere and imposing base pressure on this "column"**
  - **Allows imposition of lateral pressure gradient from WRF**

$$\Delta P = P_{WRF,G} - P_h = P_{WRF.G} - P_b\left[1 - \frac{1}{\kappa}\frac{g}{RT_b}(z - z_b)\right]^{\kappa}$$

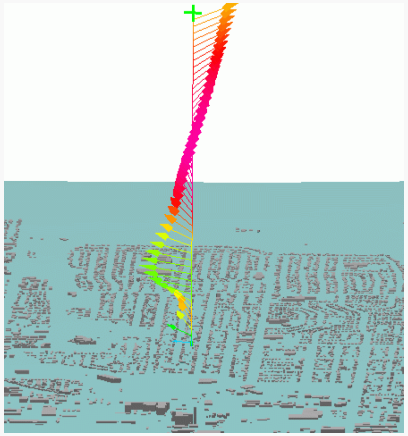
- **Turbulence Field:**
  - **Directly use TKE from the MYJ model ("TKE_MYJ")**
  - **Compute TKE dissipation rate using TKE and momentum diffusion coefficient ("AKM_M")**

$$\varepsilon = \rho C_\mu k^2 / \mu_t$$

Lateral, WRF Imposed Pressure Gradient



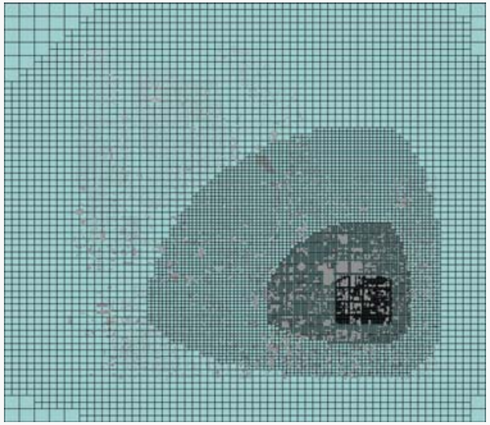
Flow Turning



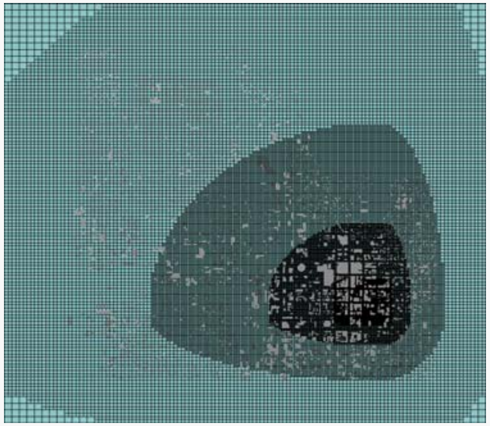
CFD-Urban Model Runs

- Unsteady Mode:
  - Solve Mass, Momentum, Turbulence Model Equations, and Transport Equation unsteady
- Quasi-Steady Mode:
  - Use libraries of equilibrium wind fields computed at different times: 15 minute intervals from WRF output
- Downscale Data Transfer (Boundary Conditions)
  - Isolated Met Station Input: Raging Waters (baseline)
  - WRF/Noah: 15 Minute Intervals
  - WRF/Noah/UCM: 15 Minute Intervals
- Cartesian Adaptive/Prismatic Grids
  - Quadtree in (x,y), Extruded in z
  - High resolution where needed with high grid quality
  - Coarse: 20 to 200 m
  - Fine: 10 to 100 m
  - 8.4 x 7.4 Km Domain

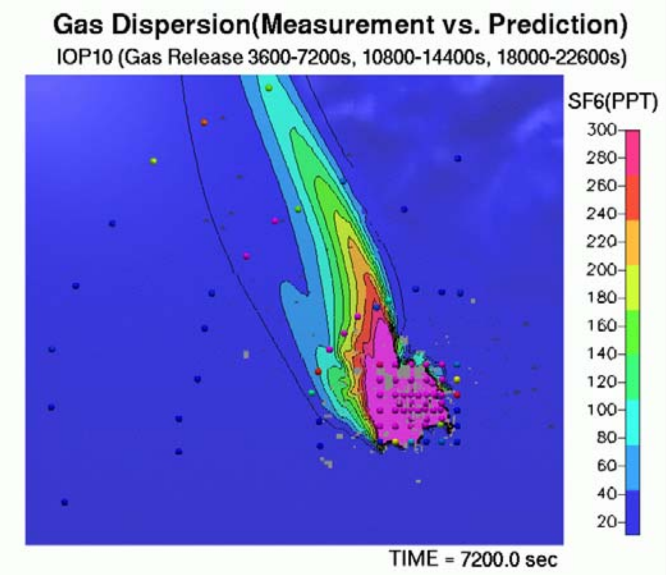
Coarse: 325,000 cells

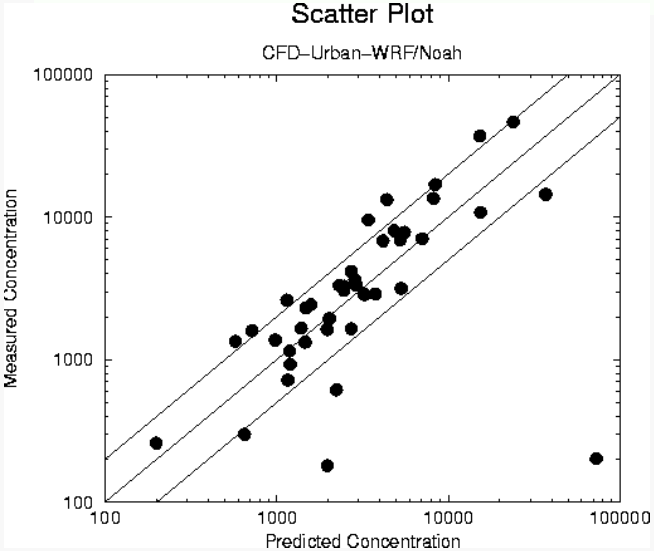


Fine: 1,300,000 cells



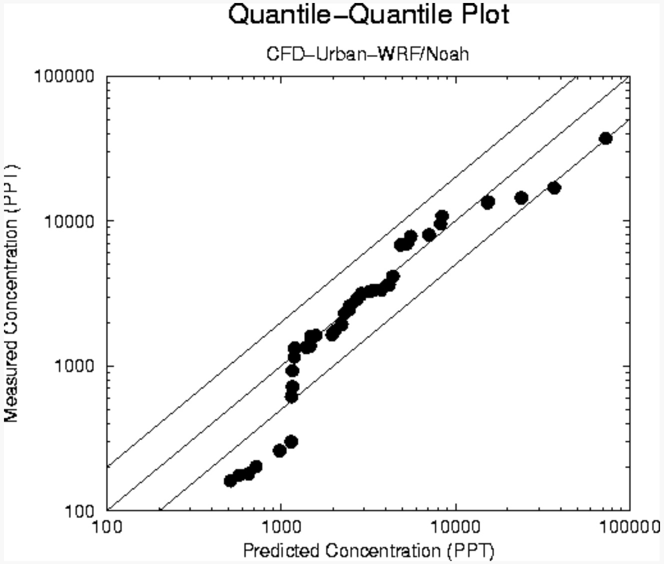
WRF/Noah Downscale: Quasi-Steady, Coarse





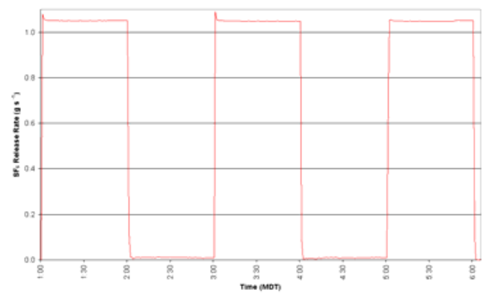
Above-to-Ground Level Shear and Plume Travel Direction Changes with Time

	Near Source	R2	R3	R4	All
FB	-0.77	0.4	0.8	0.8	-0.76
NMSE	34.36	1	2.3	1.8	53.7
MG	0.74	1.6	2	2.1	1.04
FAC2	0.57	0.4	0.4	0.5	0.51

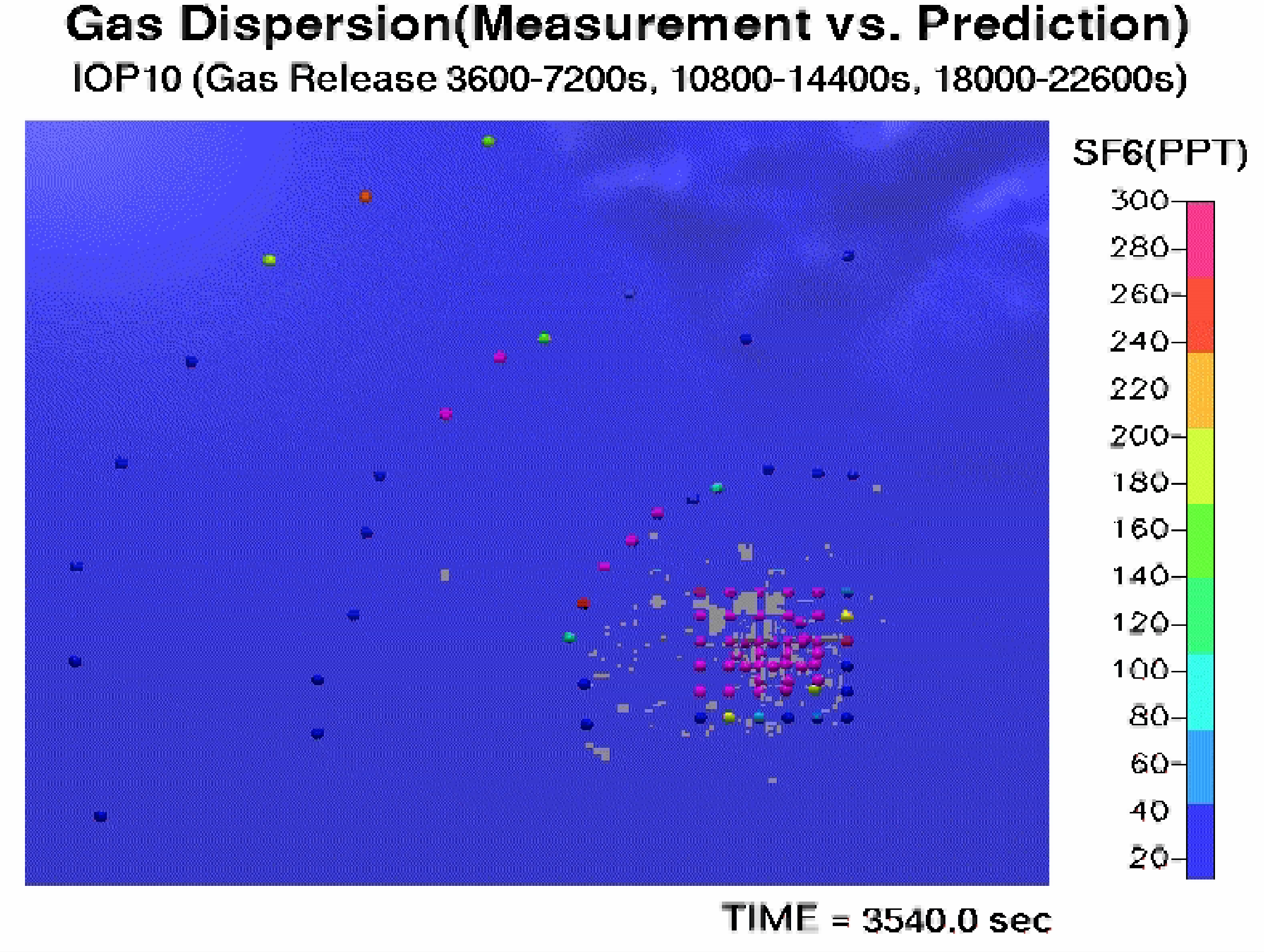


Sample of Results: IOP 10 Urban 2000

- **Entire IOP 10**
    - **3 Releases/Pauses**



- **WRF Data for BC**
- **Quasi-steady approach:**
    - **Wind/Turbulence fields at 15 minute intervals**
    - **Unsteady T&D using Unified Frozen Hydro Solver**



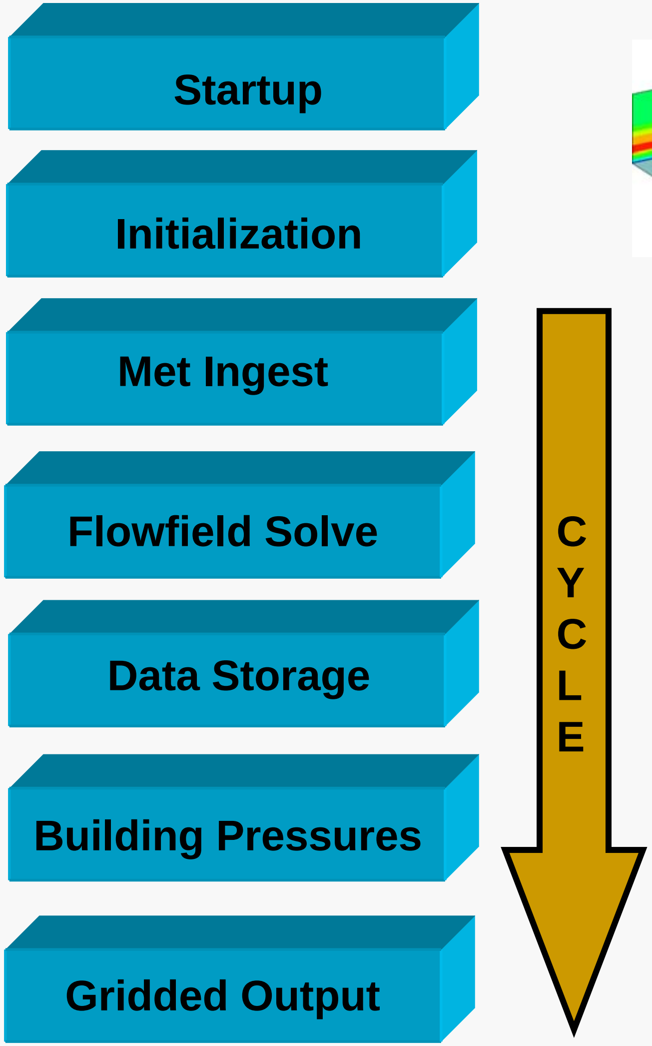
Summary of Results: IOP 10 Urban 2000

- Three sets of calculations:
    - Raging Waters Input: Use sounding data (single sounding) at all boundary faces
    - WRF Downscale Data Transfer: Unsteady Flow, Turbulence and Contaminant
    - WRF Downscale Data Transfer : Quasi-Steady Flow (“Wind Library”), Frozen Hydro Contaminant Transport

	Near Source	R2	R3	R4	All
FAC2: RW	0.12	0.17	0.36	0.38	0.18
FAC2: Unsteady	0.08	0.17	0.36	0.38	0.16
FAC2: Quasi-Unsteady	0.57	0.42	0.36	0.5	0.51
MG: RW	25.42	14.11	4.58	5.06	15.83
MG: Unsteady	15.89	11.64	4.77	5.679	11.69
MG: Quasi-Unsteady	0.74	1.59	1.96	2.05	1.04

- Quasi-Steady approach appears to be best mode of operation:
    - Steady-state wind/turbulence fields at set intervals in time using WRF data as boundary conditions: Library of Wind Fields
    - Use Unified Frozen Hydrodynamics Approach for T&D
- Unsteady flow/turbulence/transport: Time step restrictions
    - Too costly for accuracy or inaccurate because time step is too big

Pentagon Shield CFD Component: Operational Concept



Startup

Initialization

Met Ingest

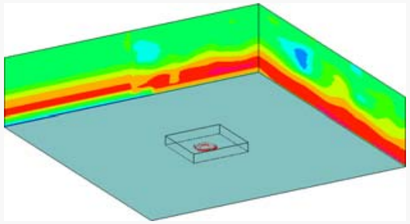
Flowfield Solve

Data Storage

Building Pressures

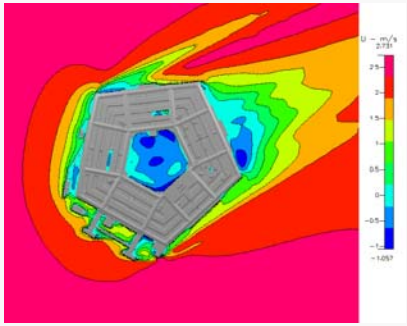
Gridded Output

CYCLE



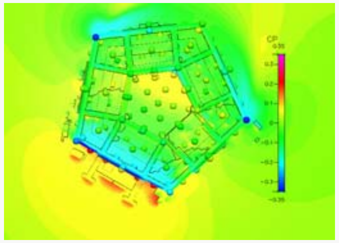
Ingest Gridded Met Data Set

Apply as Boundary Conditions



Flow+Turbulence

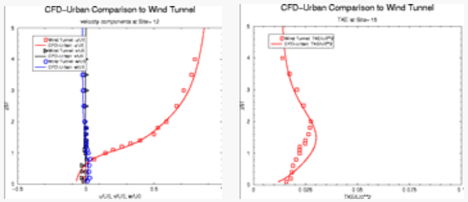
Solve



Building

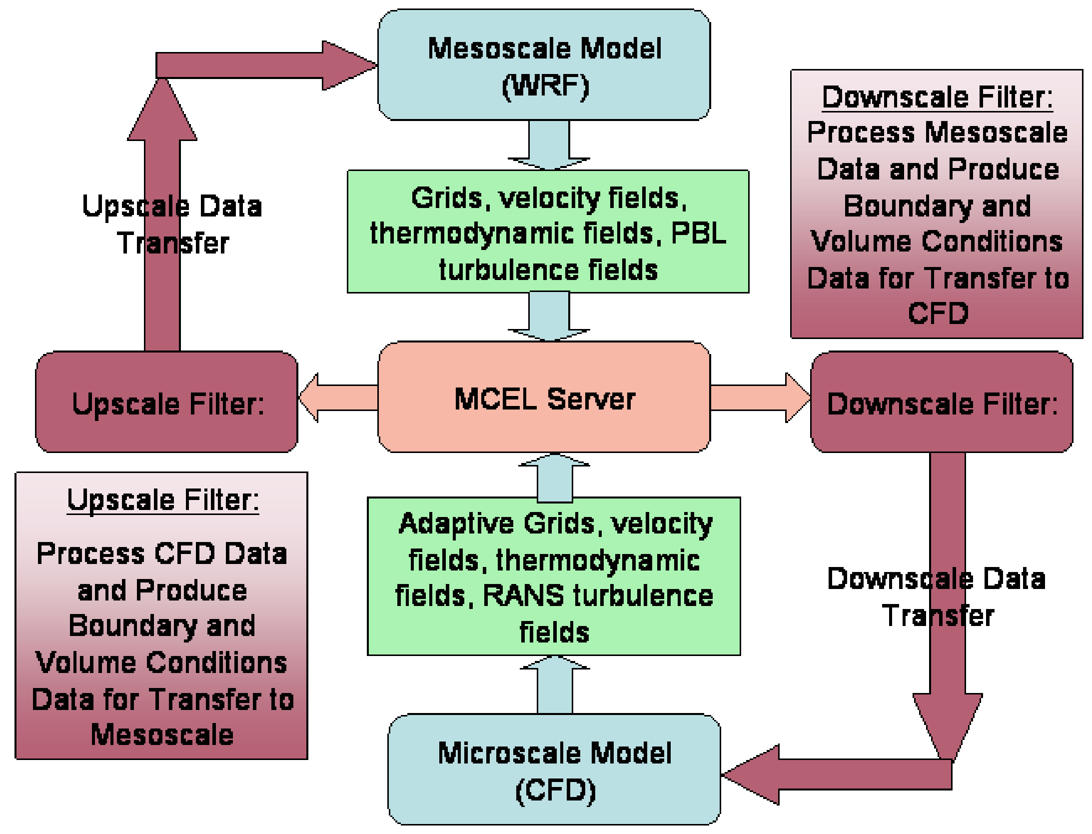
Pressures

Velocity and
Turbulence
Fields



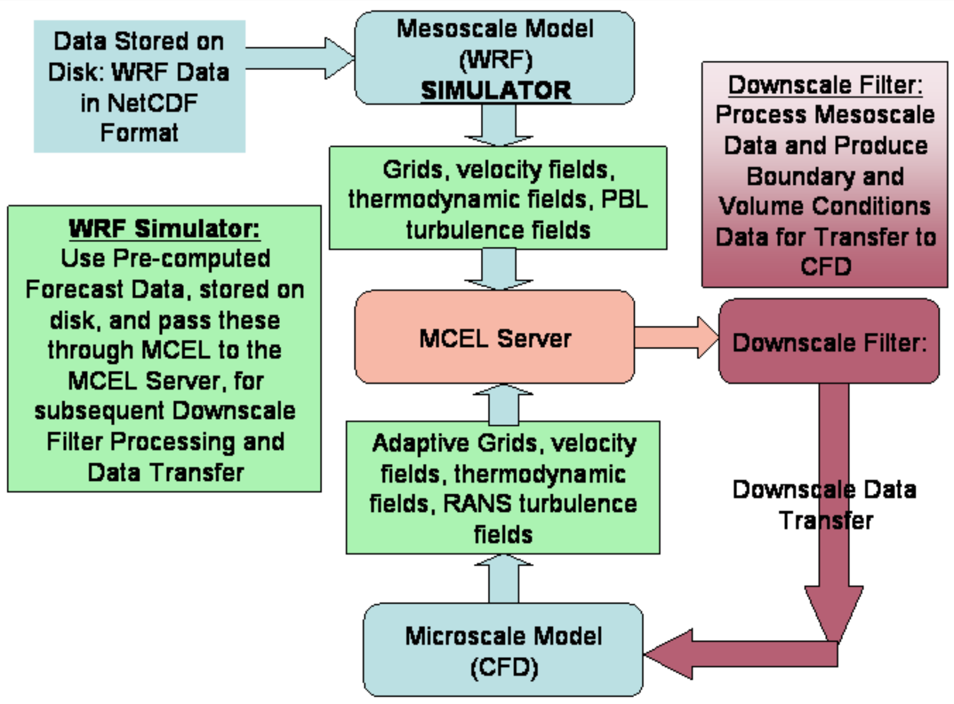
“Merged” CFD and Mesoscale Model Concept

- Model Coupling Environmental Library (MCEL) to couple:
  - Weather Research and Forecasting (WRF) Model
  - Specialized Urban CFD Model
- MCEL is a dataflow based model:
  - Models send data to the MCEL Server
  - Filters “Pull” data from servers and manipulate it
- MCEL is based upon using CORBA:
  - Client/Server design allows all components to operate relatively independently
  - Heterogeneous environments



"Merged" CFD and Mesoscale Model Concept

- **Downscale Filter:**
  - **Pulls "WRF" data from the MCEL server and "downscales" the data for use as applying boundary conditions to the CFD solver**
- **"WRF Simulator": Simulates WRF being in the loop and pushes data to the server using WRF native data formats (netCDF)**
  - **MCEL data caching capability being investigated for this purpose**


Data Stored on Disk: WRF Data in NetCDF Format

Mesoscale Model (WRF) **SIMULATOR**

Grids, velocity fields, thermodynamic fields, PBL turbulence fields

WRF Simulator: Use Pre-computed Forecast Data, stored on disk, and pass these through MCEL to the MCEL Server, for subsequent Downscale Filter Processing and Data Transfer

Downscale Filter: Process Mesoscale Data and Produce Boundary and Volume Conditions Data for Transfer to CFD

MCEL Server

Downscale Filter:

Adaptive Grids, velocity fields, thermodynamic fields, RANS turbulence fields

Downscale Data Transfer

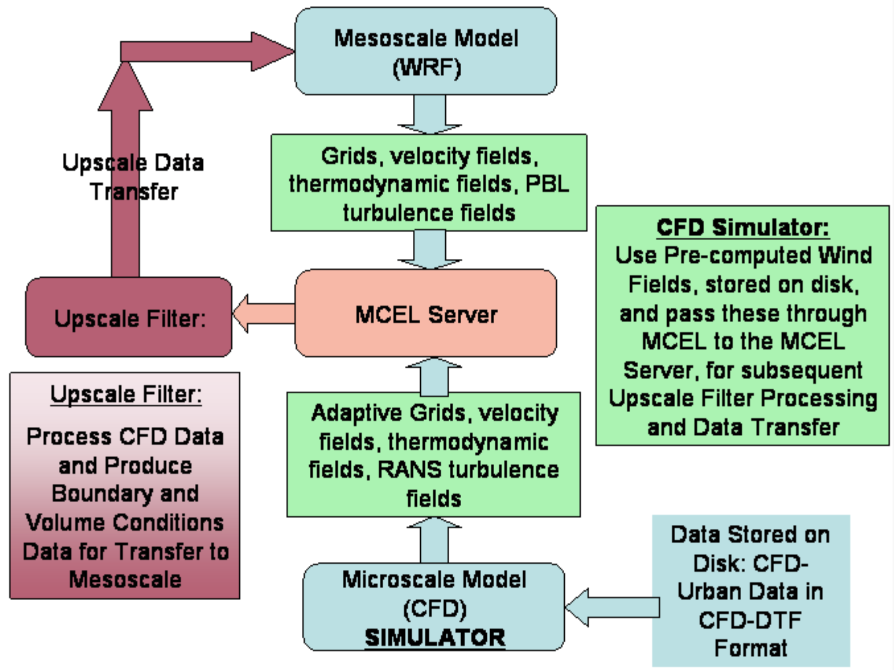
Microscale Model (CFD)

Downscaling:

- **Interpolate WRF velocity and turbulence fields onto CFD mesh**
- **Downscale "filter" pulls MCEL data from server, interpolates onto CFD mesh**
- **Uses "sub-cube" of finest nest**

“Merged” CFD and Mesoscale Model Concept

- **Upscale Filter:**
    - **Pulls “CFD” data from the MCEL server and “upscales” the data for use in WRF**
    - **Debating what to upscale, but consensus appears to be volumetric data**
- **“CFD Simulator”:Simulates CFD being in the loop and pushes data to the server using CFD native data formats (CFD-DTF)**
    - **MCEL data caching capability being investigated for this purpose**



Mesoscale Model (WRF)

Grids, velocity fields, thermodynamic fields, PBL turbulence fields

Upscale Data Transfer

MCEL Server

Upscale Filter:

CFD Simulator: Use Pre-computed Wind Fields, stored on disk, and pass these through MCEL to the MCEL Server, for subsequent Upscale Filter Processing and Data Transfer

Upscale Filter: Process CFD Data and Produce Boundary and Volume Conditions Data for Transfer to Mesoscale

Adaptive Grids, velocity fields, thermodynamic fields, RANS turbulence fields

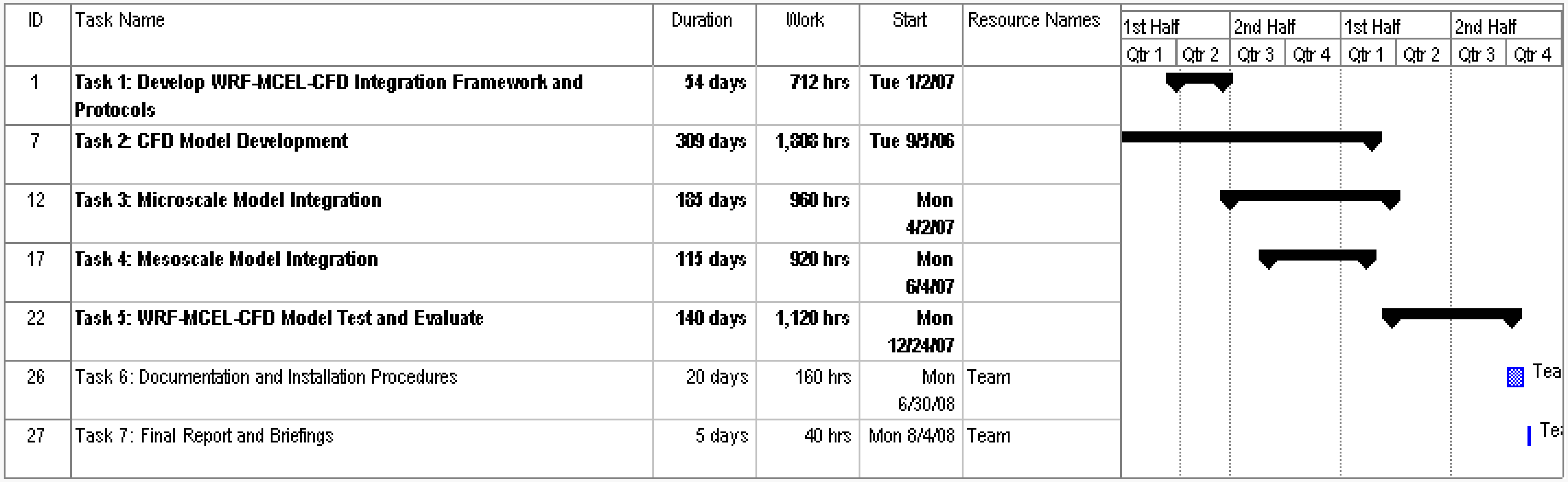
Microscale Model (CFD) **SIMULATOR**

Data Stored on Disk: CFD-Urban Data in CFD-DTF Format

Upscaling:

- **Compute momentum flux-integral for all cells contained within (coarser) WRF cells**
- **“Decimation” produces body force tendency, replaces drag terms from urban canopy model**
- **Uses “sub-cube” of finest nest**

Schedule and Work Plan



ID	Task Name	Duration	Work	Start	Resource Names
1	**Task 1: Develop WRF-MCEL-CFD Integration Framework and Protocols**	**54 days**	**712 hrs**	**Tue 1/2/07**	
7	**Task 2: CFD Model Development**	**309 days**	**1,808 hrs**	**Tue 9/5/06**	
12	**Task 3: Microscale Model Integration**	**185 days**	**960 hrs**	**Mon 4/2/07**	
17	**Task 4: Mesoscale Model Integration**	**115 days**	**920 hrs**	**Mon 6/4/07**	
22	**Task 5: WRF-MCEL-CFD Model Test and Evaluate**	**140 days**	**1,120 hrs**	**Mon 12/24/07**	
26	Task 6: Documentation and Installation Procedures	20 days	160 hrs	Mon 6/30/08	Team
27	Task 7: Final Report and Briefings	5 days	40 hrs	Mon 8/4/08	Team

Task 1: Upscale/Downscale Data and Protocols are being defined (delays getting all contracts/subcontracts in place)

Task 2: New, specialized solver and system under development

Task 3: Not yet begun

Task 4: Not yet begun

Task 5: Not yet begun

Task 6: Not yet begun

Specialized CFD Solver and Related Development

- **Upgrade mesh generation, numerics and parallel processing software to permit faster processing rates and higher fidelity physics:**
  - **Mesh Generation:**
    - **Cartesian Adaptive grids combined with Virtual Cell Embedding and cell-based Porosity for sub-cell resolution of features**
  - **Numerics:**
    - **Low-Mach Number Pre-Conditioned, Reynolds-Averaged Navier-Stokes Equations**
    - **Coupled mass+momentum+energy more suitable for atmospheric flows**
  - **Parallel Processing Software**
    - **Maximal use of the PETSc (Portable Extensible Toolkit for Scientific Computing) library**
    - **System of Non-Linear Equation Solvers (SNES) framework**
      - **Newton-Krylov-Schwarz parallel, implicit**

“Feed the PETSc engine”

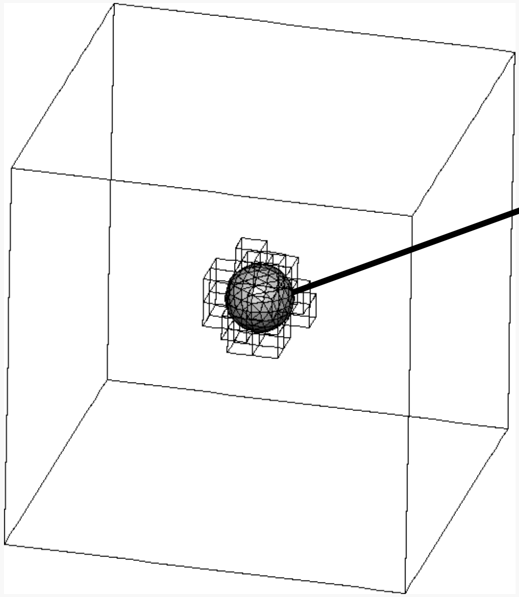
Technical Progress to date: AMR_VCE

- Adaptive Mesh Refinement Virtual Cell Embedding: AMR_VCE
- Solution adaptive Cartesian mesh using hierarchical (octree) system
    - Based upon HAMR classes and techniques
- Triangulated surface queries and tools using the GNU Triangulated Surface Library (GTS)
- Basic Algorithm:
    - Refine mesh where cells are “cut” by body (in/out) until a given level of refinement
    - Take each cell and “virtually” refine it near the boundary until a given level of (much finer) refinement
    - Use the geometric information of the adaptively refined “virtually embedded cell” to obtain approximations to the boundary to be represented in the larger “parent” cell
    - Insure Geometric Conservation Law consistency during construction of the cut cell centroids, integration points etc.
    - An “Adaptive Mesh Refinement” refinement of VCE [Landsbury, Boris]

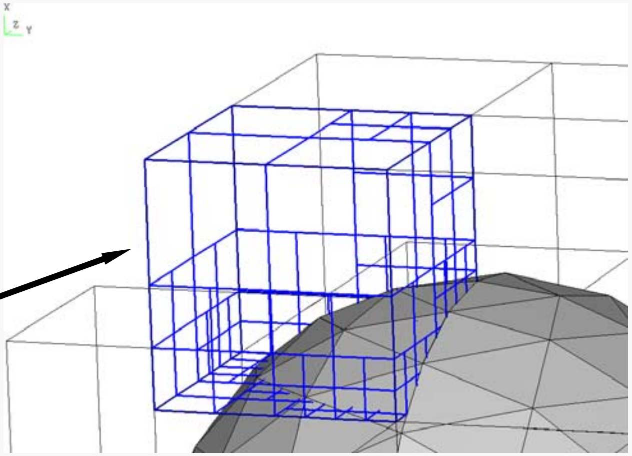
Technical Progress to date: AMR_VCE

- **Example: Sphere (represented via STL)**

VCE Cells Boundaries



Single AMR_VCE Cell (blue)

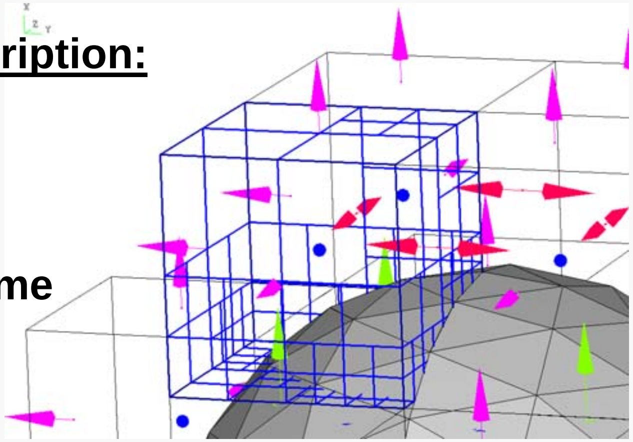


<u>“Cut” Cell Geometric Description:</u>

Cut Cartesian Faces (CCF)

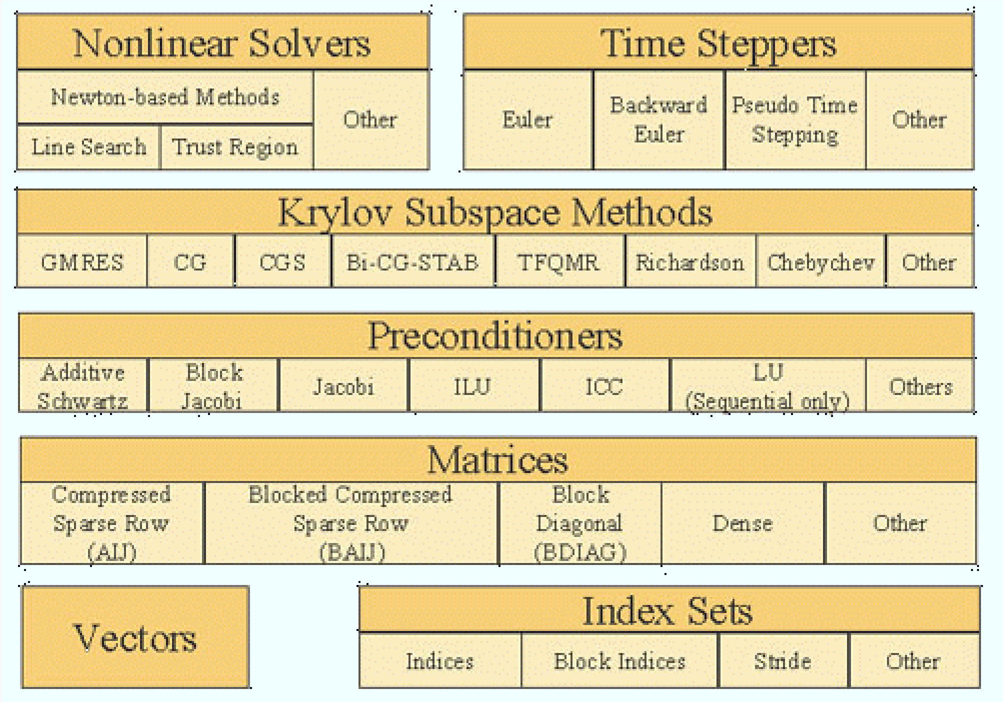
Cut Face

Cut Cell Centroid and Volume



Technical Progress to date: Flow Solver Technology

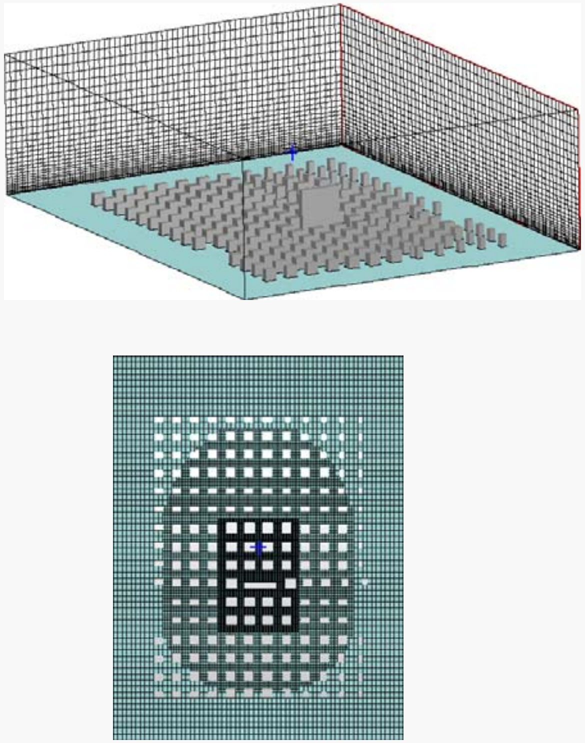
- **Evaluation and Testing of PETSc**
- **Parallel Extensible Toolkit for Scientific Computation (PETSc)**
  - **Developed by Argonne National Labs for parallel solution of sparse matrices encountered in scientific computing.**
  - **Becoming a very widely used application**

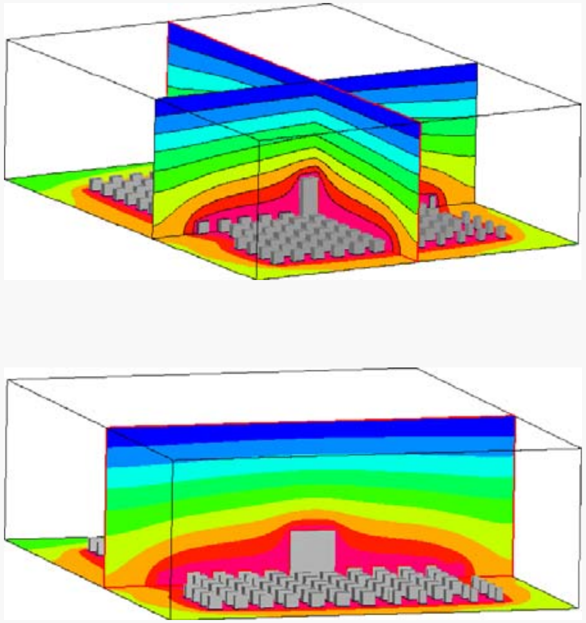


- **Parallel computing framework using basic “classes” supported in the PETSc library**
- **Provides large variety of Krylov-based solvers and matrix preconditioners, all in parallel.**
- **Proven to be very efficient and scalable.**
- **Choices of use:**
  - **Develop as usual, retrofit PETSc**
  - **Design from start to use PETSc**

Technical Progress to date: Flow Solver Technology

- “Iterative” Approach to developing the PETSc-specialized flow solver:
  - Laplace Equation solver: Serial, Parallel
  - Euler Equation Solver: Parallel, low-Mach Number Preconditioned
  - RANS Solver: Add viscous terms and 2-equation Turbulence Model ($k\varepsilon$ or MYJ)





Boundary Conditions:

- u=1 on Buildings
- u=0 on “Sky” boundary
- Zero gradient on all others
- 380,000 unknowns
- 2.4M matrix elements

Flow Solver Technology: Low Mach Preconditioning

- **Low-Mach Number Preconditioning Approach: [Weiss, et al., 1995, Merkle, et al., 1996]**
  - **Fully-coupled mass, momentum and energy equations, “Compressible” formulation**
  - **Preconditioning removes stiffness of equations as Mach number approach zero**

$$\frac{\partial}{\partial t}\iiint W dV + \iint [F - G] \bullet dA = 0$$

Conservation Law Form, Conserved Variables

$$\kappa(\lambda) \approx \frac{\lambda_{\max} - \lambda_{\min}}{\left(\lambda_{\max} + \lambda_{\min}\right)/2}$$

Condition number of hyperbolic system is directly related to the eigenvalues

$$\left(K\frac{\partial W}{\partial Q}\right)\frac{\partial}{\partial t}\iiint Q dV + K\iint [F - G] \bullet dA = 0$$

Primitive Variables, Pre-multiply by preconditioning matrix, K (not in conservative form now…)

$$K\frac{\partial W}{\partial Q} = \begin{pmatrix} \rho_P & 0 & 0 & 0 & \rho_T \\ 0 & \rho & 0 & 0 & 0 \\ 0 & 0 & \rho & 0 & 0 \\ 0 & 0 & 0 & \rho & 0 \\ -1 & 0 & 0 & 0 & \rho C_P \end{pmatrix}$$

Note that terms pre-multiplying density time derivate approach zero as the Mach number approaches zero, decoupling mass from the other equations.

$$\kappa(\lambda) \approx \frac{1}{M}$$

Flow Solver Technology: Low Mach Preconditioning

$$\Gamma \frac{\partial}{\partial t}\iiint Q dV + \iint \left[F - G\right] \bullet dA = 0$$

Conservation Law Form, Primitive Variables, pre-conditioned equations

$$\Gamma = \left(K^{-1}\Gamma_{nc}\right)$$

$$= \begin{pmatrix} \Theta & 0 & 0 & 0 & \rho_T \\ \Theta v_x & \rho & 0 & 0 & \rho_T v_x \\ \Theta v_y & 0 & \rho & 0 & \rho_T v_y \\ \Theta v_z & 0 & 0 & \rho & \rho_T v_z \\ \Theta H - 1 & \rho v_x & \rho v_y & \rho v_z & \rho_T H + \rho C_P \end{pmatrix}$$

Low-Mach Number Preconditioning replaces "true" EOS derivatives with modified forms that removes the stiffness of the equations:

$$\Theta = \left(\frac{1}{U_r^2} - \frac{\rho_T}{\rho C_P}\right) \qquad U_r = \begin{cases} \varepsilon c, & if \quad |v| < \varepsilon c \\ |v|, & if \quad \varepsilon c < |v| < c \\ c, & if \quad |v| > c \end{cases}$$

$$u = v.\hat{n} \quad u' = u(1-\alpha) \quad \alpha = \left(1 - \beta U_r^2\right)/2 \quad c' = \sqrt{\alpha^2 u^2 + U_r^2}$$

- **System of equations now remains well conditioned at all speeds and for all EOS**

$$\lambda\left(\Gamma^{-1}\frac{\partial F}{\partial Q}\right) = u, u, u, u' + c', u' - c'$$

$$\kappa(\lambda) \approx \frac{\sqrt{\alpha^2 + (U_r/u)^2}}{(1-\alpha)}$$

- **Compressible** $\kappa(\lambda) \approx \frac{1}{M}$
- **Compressible, Low Mach** $\kappa(\lambda) \approx 1$
- **Incompressible** $\kappa(\lambda) \approx 1$

Flow Solver Technology: Low Mach Preconditioning

$$\Gamma \frac{\partial \tilde{Q}}{\partial t}\Delta V + \sum_{faces}\left[\tilde{F} - \tilde{G}\right]A_{face} = 0$$

$$F = \frac{1}{2}\left(F_L + F_R\right) - \frac{1}{2}\left|A\right|\Delta W$$

$$\left|A\right|\Delta W \cong A\Delta W = \Gamma\left|A_\Gamma\right|\Delta Q$$

Discrete, Conservative Form, Primitive Variables, Pre-conditioned System

Upwind (Hyperbolic Systems) Form

Roe's FDS or Central differencing with 4-th order dissipation using preconditioned hyperbolic system Eigenvalues and Eigenvectors

("Speed" vs "Accuracy": Who will win?)

- **Use Newton-Krylov-Schwarz Approach to solve the discrete, preconditioned equations:**
  - **Proven scalability and high-performance using the PETSc suite**
  - **Similar techniques as PETSc-FUN3D, which won the 1999 Gordon Bell Award**

$$\left[\frac{\Gamma}{\Delta t} - \frac{\partial R}{\partial Q}\right]\Delta Q = R(Q)$$

$$CFL^N = CFL^{N-1}\left(\frac{\left\|f^{N-1}\right\|}{\left\|f^{N}\right\|}\right)^{\beta}$$

Technical Progress to date: Flow Solver Technology

- Developing flow solver from the ground up to be PETSc compliant
- Using lessons learned from many different unstructured flow solvers, parallel processing studies:
  - Interleaved data structures, ordering of unknowns and visitation order (coloring)
  - Reduction in cache misses, communication patterns, line-search strategies…
- Use the PETSc SNES to solve the parallel system of equations:
  - Provide 2 functions called by SNES solver:
    - Residual evaluation and Jacobian evaluation

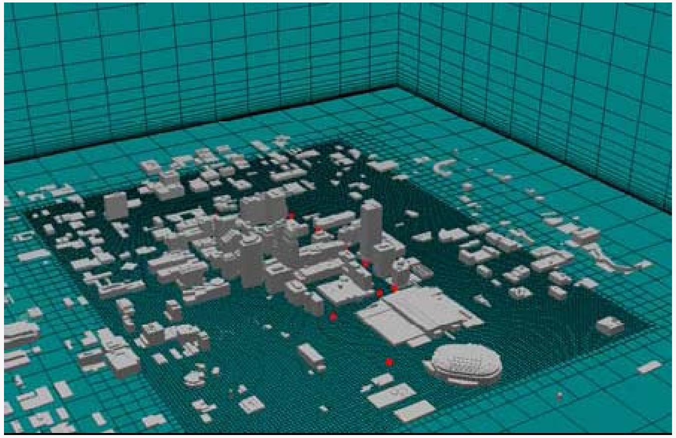
```
SNESCreate(MPI_COMM_WORLD,&snes);

SNESSetType(snes,"ls");

SNESSetFunction(snes,pData-
>rhs,formFunction,PETSC_NULL);

MatCreateMPIBAIJ(PETSC_COMM_WORLD,bs,m,n,M,
N,&d_nnz,d_nz,o_nnz,o_nz, &J)

SNESSetJacobian(snes,J,J,formJacobian,PETSC
_NULL);
```

```
SNESGetKSP(snes,&ksp);

KSPSetType(ksp,KSPBCGSL);

SNESSetTolerances(snes,1.0e-10,1.0e-
10,1.0e-40,10000,10000);

KSPSetTolerances(ksp,1.e-10,1.e-
10,1e+50,100);

SNESSolve(snes,PETSC_NULL,pData->sol);
```

FEED THE PETSc SNES SOLVER

Technical Progress to date: Flow Solver Technology

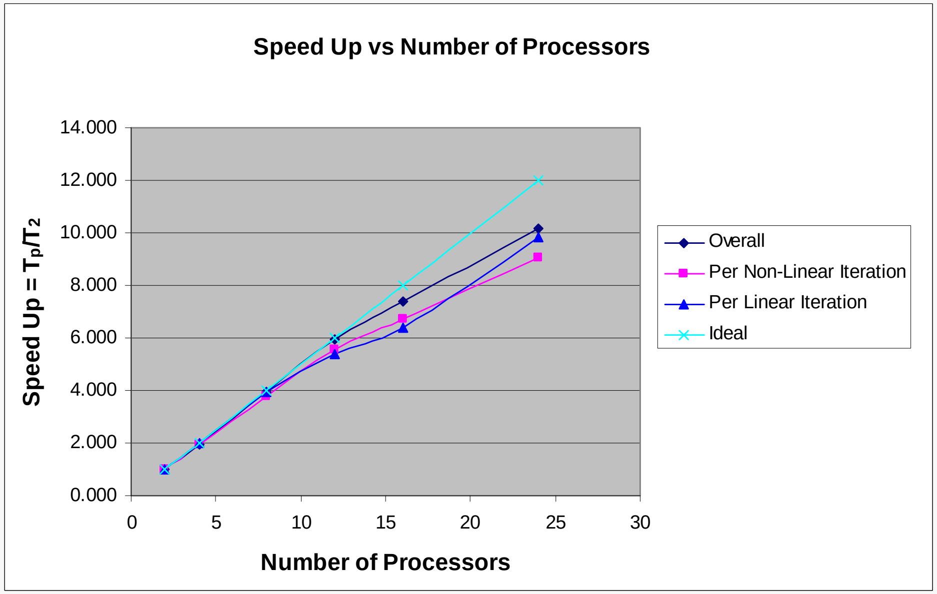
- **Scalability study: Euler equations, non-preconditioned**

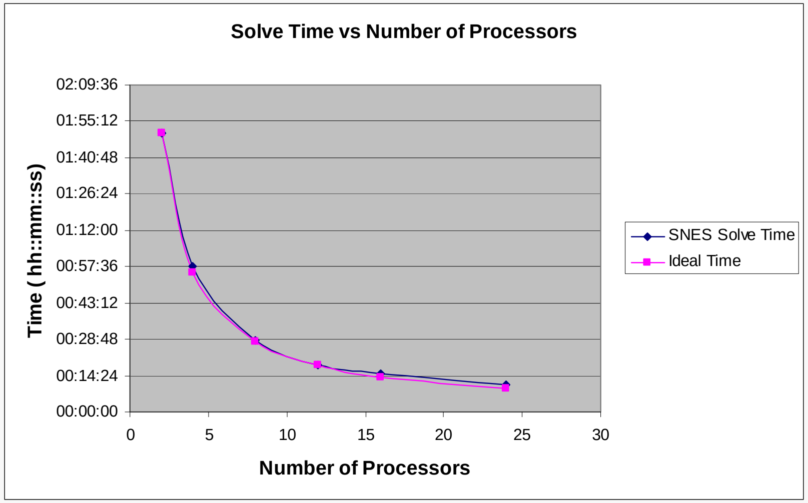


Case 1: Oklahoma City CBD Grid generated by CFD-Urban

- **175,000 cells**

•Linear Solver: BiCGSTAB*(l)*, Preconditioner: Block Jacobi, ILU on each block.





Tail-off after 12 processors due to using dual processors(!)

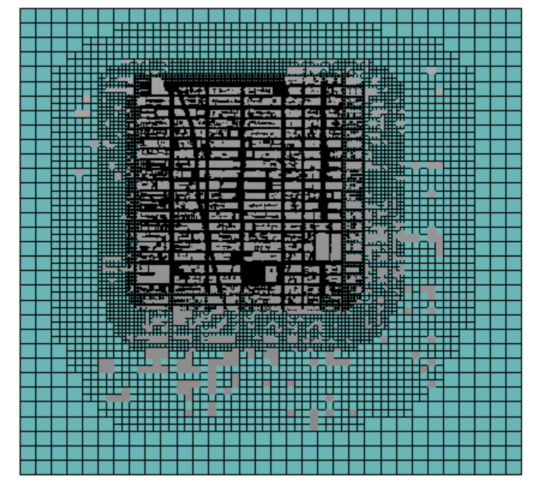
Technical Progress to date: Flow Solver Technology

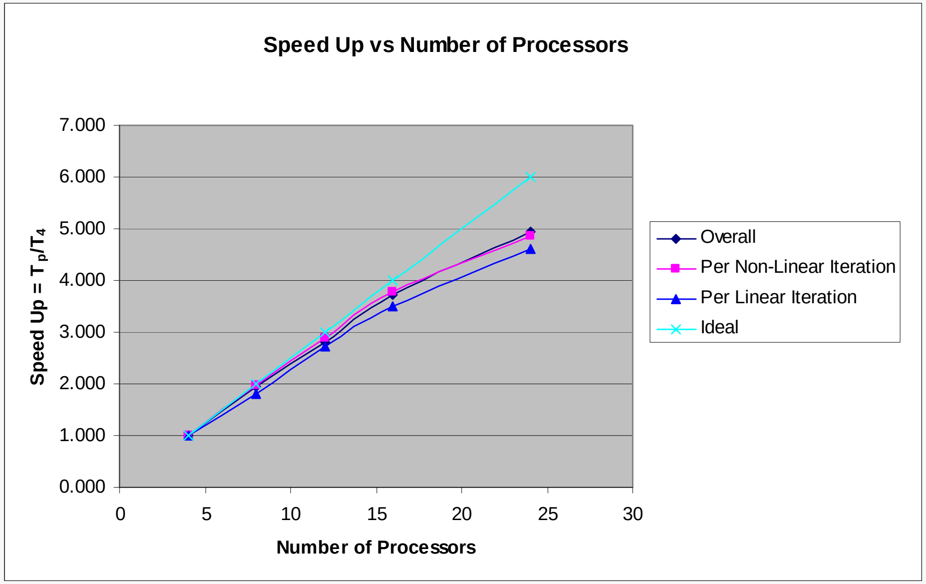
- **Scalability study: Euler equations, non-preconditioned**

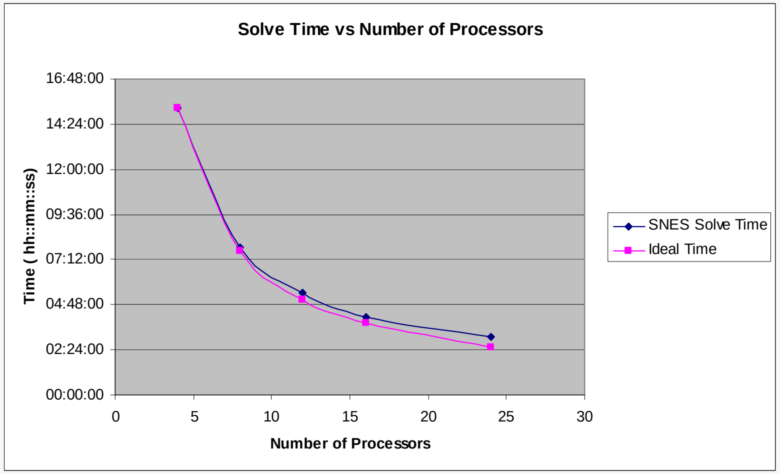
Case 2: Midtown Manhattan Grid generated by CFD-Urban

- **1,460,000 cells**

- **Linear Solver: BiCGSTAB*(l)*, Preconditioner: Block Jacobi, ILU on each block.**



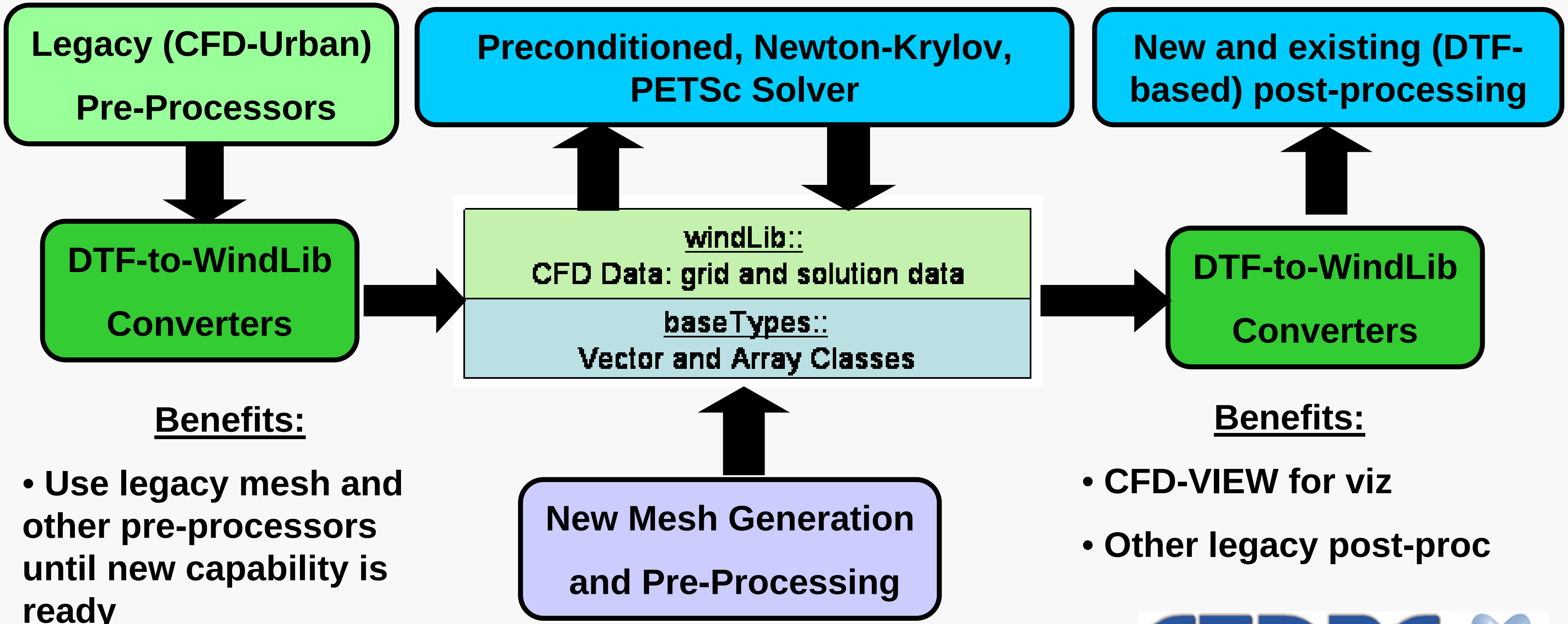




Tail-off after 12 processors due to using dual processors(!)

Wind Library Database API and Classes

- C++ classes/namespaces devoted to storage and access of grids and solution data
- Move away from ESI (ex-CFDRC) proprietary CFD-DTF and related libraries
- baseTypes:: derived from std namespace vector<T>
- windLib:: hierarchical data structures stored using simple strings and disk access



Legacy (CFD-Urban) Pre-Processors

DTF-to-WindLib Converters

Preconditioned, Newton-Krylov, PETSc Solver

windLib::
CFD Data: grid and solution data

baseTypes::
Vector and Array Classes

New Mesh Generation and Pre-Processing

New and existing (DTF-based) post-processing

DTF-to-WindLib Converters

Benefits:

- Use legacy mesh and other pre-processors until new capability is ready

Benefits:

- CFD-VIEW for viz
- Other legacy post-proc

Conclusions and Plans

- **Beginning to work with MCEL, downscale/upscale filters**
- **Flow solver development is proceeding according to plan**
- **If all goes according to plan: Will be coupled this time next year**