Recent Developments in ConfinedBlast Web-Application

SLIDES ONLY

NO SCRIPT PROVIDED

Serdar Astarlioglu, Ph.D.
Safety Engineer
Explosives Safety Office

IESSE 25
Phoenix, AZ
January 21-23, 2026

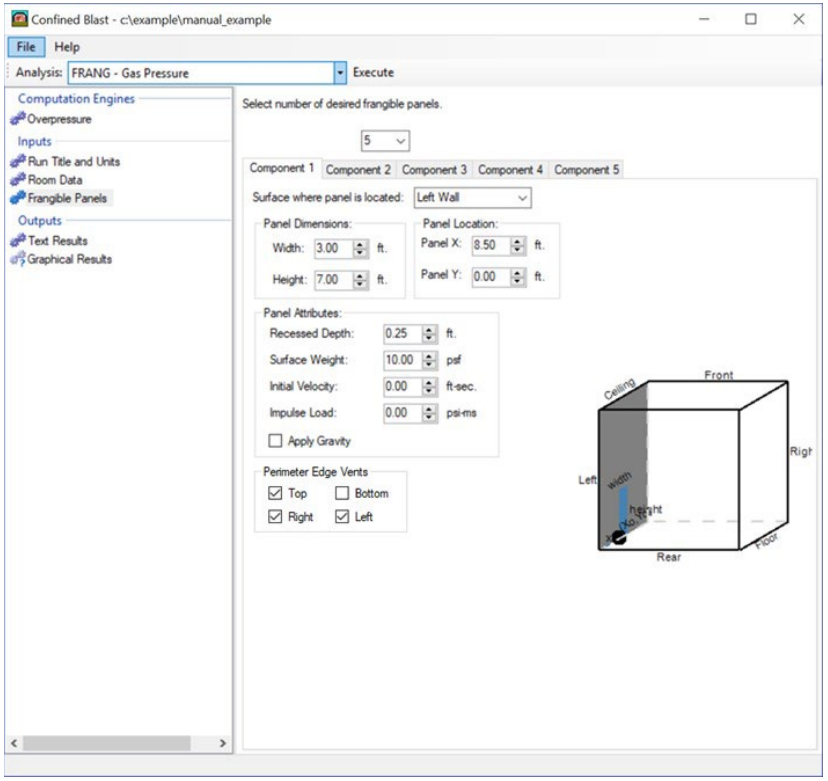
Introduction

ConfinedBlast is a software tool approved by Explosives Safety Board for performing blast and debris calculations. It consists of three modules:

- SHOCK: calculates pressures and impulses on walls from an explosion during the shock phase.
- FRANG: calculates pressures and impulses on walls of a confined cubicle during the gas phase.
- MUDEMIMP: Determines downrange hazardous distance and dispersion from debris breakup

Introduction (continued)

- ConfinedBlast is a standalone Windows Desktop application that supports solvers written in FORTRAN and user-interface written using Visual Basic (Microsoft .NET Framework)
- Latest version is ConfinedBlast v5.0.1.0 with release date Aug 18, 2022



Software Issues

- Current IT restrictions prevent ConfinedBlast from being installed in most DoW computers
- New operating systems and processor architectures require different versions of the software
  - no MacOS, Linux OS,
  - ARM processor support (recent non-Intel Windows laptops)
- The source code for some of the solvers is not available (i.e., MUDEMIMP uses DLL compiled in early 2000s)

Solver Issues

- Both SHOCK and FRANG support triangular ramp-down pressure-time histories (peak pressure and impulse). This leads to more conservative design than needed (we already have factors for conservatism in the charge weight)
- In SHOCK, the number and locations of the sampling points/gauges are fixed. The average pressure and impulse calculation does not take shock arrival time into account. Issues with large walls, near-field charges

Solver Issues (continued)

- FRANG does not include effects of rise time in gas pressures and may overestimate peak gas pressure when the volume is not fully enclosed. Instantaneous rise time leads to more conservative blast loading
- Existing MUDEMIMP DLL is a dead end, without reliable source code there is no potential to make updates to the algorithms to reflect the state-of-the-art in secondary debris methodologies (i.e., account for larger debris sizes, updates to roll equations)

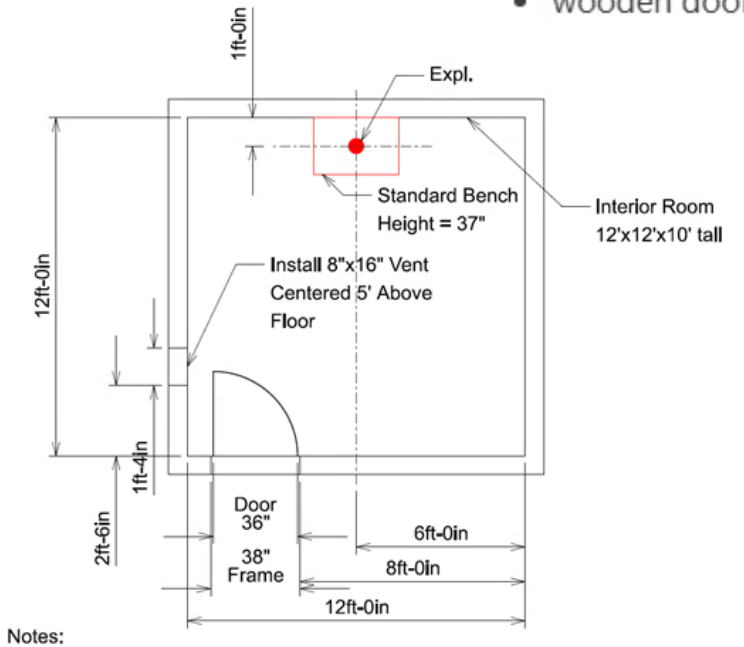
Current Progress

- Replaced SHOCK module with shock solver from BlastX v9
- Replaced FRANG with new fast-running gas solver algorithm developed by Dr. Oswald.
- Developed brand new code for MUDEMIMP from scratch, rather than fixing and updating the old unreliable source code.
- The end product will be a web-based application
- ERDC/ITL is tasked with developing the web user interface
- EXWC is tasked with comparing new solvers with legacy ones

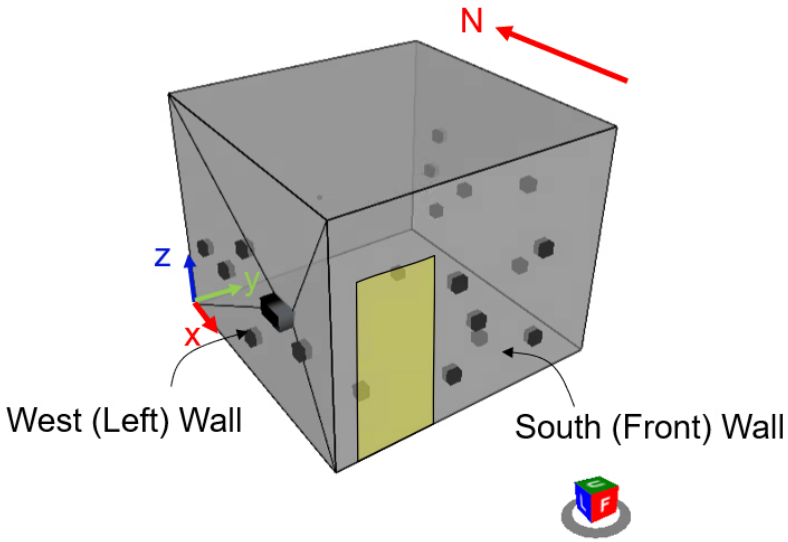
Model Setup

Test setup:

- 12-ft x 12-ft x 10-ft room
- 16-in x 8-in vent on west (left) wall
- 50-gm charge is 1-ft offset from north (back) wall
- wooden door is on south (front) wall



Notes:
Doors: Standard residential exterior wooden doors, 36"w x 80"t
Explosives: Held 12" above table top by cardboard tube



Dimensions and Charge Configuration

```
var model = new BlastModel();
model.ReflectingSurfaces = [
    Location.Down, Location.Up,
    Location.Left, Location.Right,
    Location.Front, Location.Back];
model.Length = 12 * ft;
model.Width = 12 * ft;
model.Height = 10 * ft;
```

✓ 0.5s csharp - C# Script Code

```
model.SafetyFactor = 1.0;
model.ChargeMass = 50 * gm;
model.ChargePosition = new(1 * ft, 6 * ft, 4 * ft + 1 * inch);
```

csharp - C# Script Code

ConfinedBlast has a built-in table for over 50 explosives (TNT equivalency, heats of detonation and combustion)

Panel Types

- Opening
  - Wall location
  - Dimensions
- Closed Vent (Frangible Panel)
  - Mass
  - Initial impulse (shock impulse used)
  - Wall location
  - Position
  - Dimensions
- Area (Gauge array or point gauge)
  - Wall location
  - Position
  - Dimensions (0x0 dimension → point gauge)

Vent, Frangible Panel, and Gauges

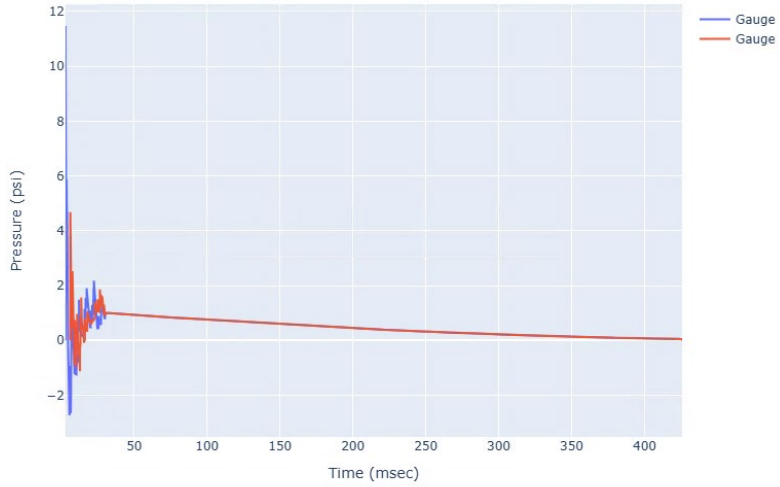
```csharp
var vent = new Panel();
vent.Type = PanelType.Opening;
vent.Location = Location.Left;
vent.Position = new(8 * ft + 2 * inch, 4 * ft + 8 * inch);
vent.Dimension = new(16 * inch, 8 * inch);
model.Panels.Add(vent);
```

csharp - C# Script Code

```csharp
var gauges = new Panel[20];
gauges[0] = Panel.CreatePointGauge(location: Location.Down, position: new(3 * ft, 3 * ft));
gauges[1] = Panel.CreatePointGauge(location: Location.Down, position: new(3 * ft, 9 * ft));
gauges[2] = Panel.CreatePointGauge(location: Location.Down, position: new(9 * ft, 3 * ft));
gauges[3] = Panel.CreatePointGauge(location: Location.Down, position: new(9 * ft, 9 * ft));
```

Results from Analysis

```
gauge index, peak pressure (psi), peak impulse (psi*msec)
0, 11.4696527738396, 186.45071898845532
1, 4.6806373918435416, 185.4268281860061
2, 11.469651640732227, 184.7884472017179
3, 4.680637108566698, 184.596645743497
4, 12.970162282369053, 185.94316960176116
5, 8.478417572403204, 186.07060366916238
6, 6.041245815479792, 186.52490523801688
7, 6.0971963914029175, 185.39069978081884
8, 3.834109302637497, 186.18437717619676
```



- ConfinedBlast combines pressure-time histories from shock and gas calculations
- Peak pressure/impulse summaries are provided
- These histories can be mapped to SDOF or FEA solvers for structural analysis

Summary

- This project aims to make ConfinedBlast usable across the DoW through transition to a web-based environment
- Updates legacy solvers (20+ years old) with state-of-the-art algorithms that are more accurate
- Web interface is still under development (Anticipating completion by Q4 FY26)
- The new framework also enables future improvements such as adding HD 1.3 solvers, SDOF modules