

Configuration-Aware Performance Analysis



Florian
Sattler



Christian
Kaltenecker



Sven
Apel

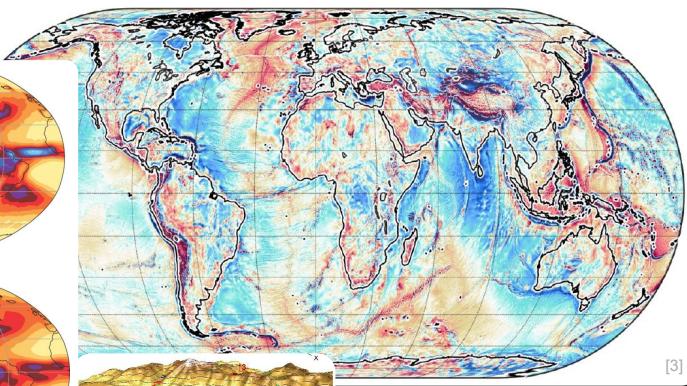
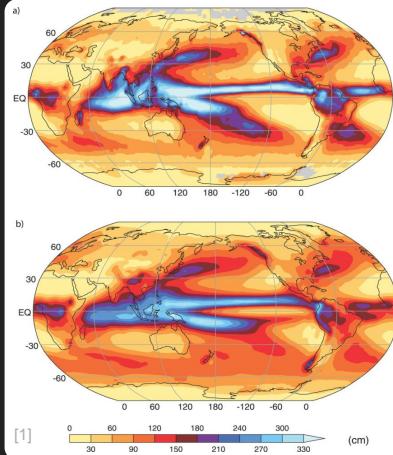


UNIVERSITÄT
DES
SAARLANDES

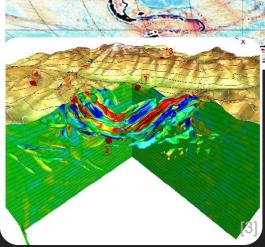


Global gravity field modelling

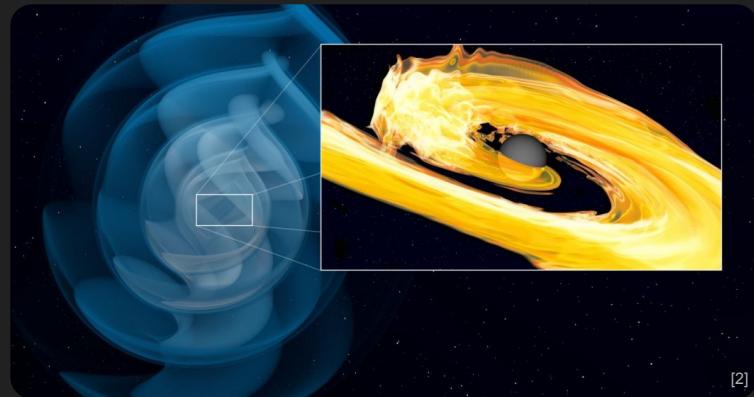
Climate models



3D seismic wave propagation

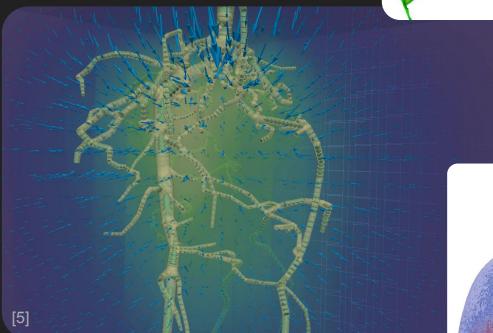
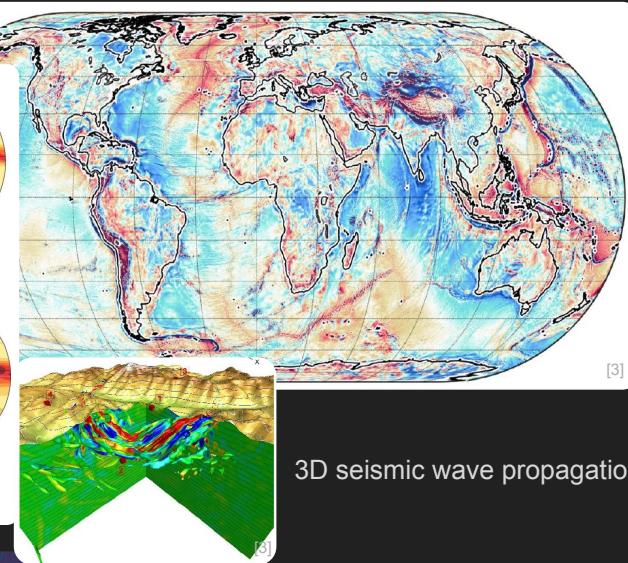
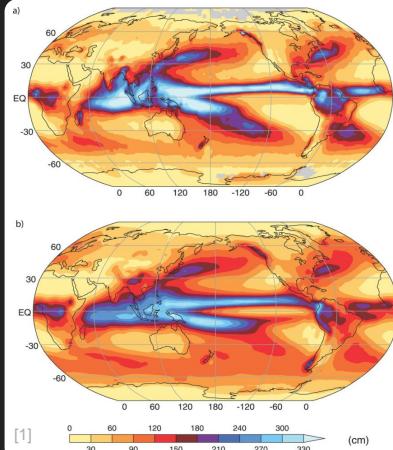


Simulation of a black-hole–neutron-star merger

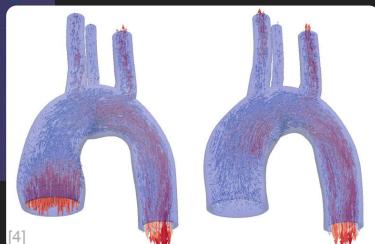


Global gravity field modelling

Climate models

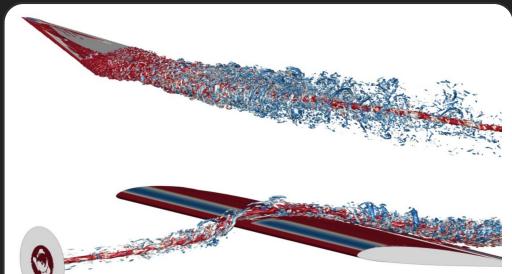
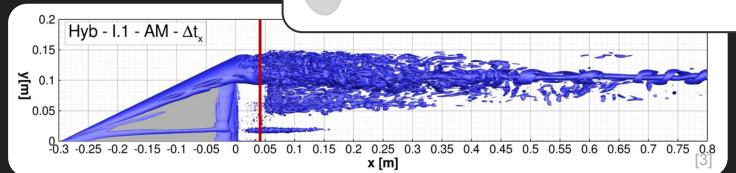
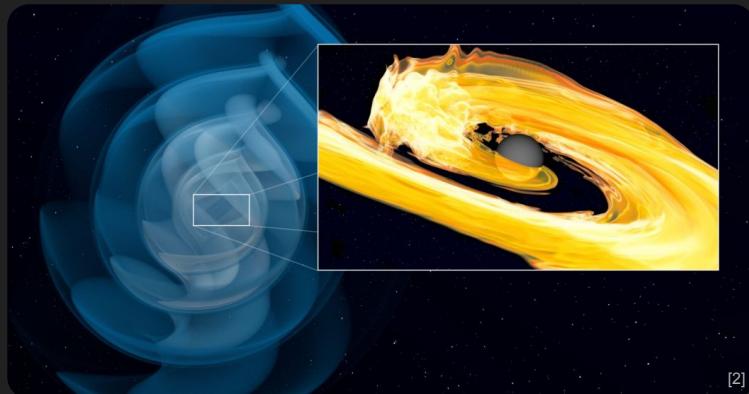


Water flow field in the soil and the root xylem.



Fluid-structure interaction of an aortic blood flow.

Simulation of a black-hole–neutron-star merger



Strake vortex simulations

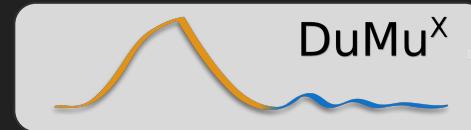
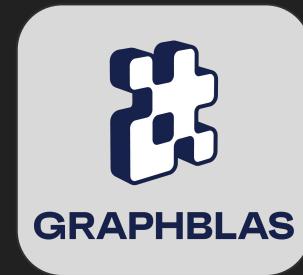


All these problems are extremely compute intensive!

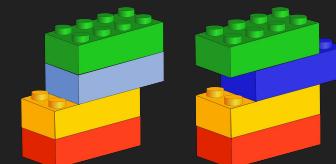


Better performance through hardware AND software.

HPC-Frameworks

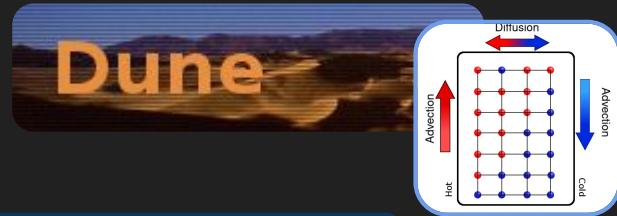


All these frameworks provide building blocks to develop and solve a variety of problems from a diverse set of scientific disciplines.



Configurability

Configurability of HPC-Frameworks



Compile-time configuration choices

```
typedef AlgorithmConfigurator< GridSelectorGridType,
    Galerkin::Enum::default_,
    Adaptivity::Enum::default_,
    //DiscreteFunctionSpaces::Enum::gausslobatto,
    //DiscreteFunctionSpaces::Enum::gausslegendre,
    DiscreteFunctionSpaces::Enum::legendre,
    //DiscreteFunctionSpaces::Enum::default_,
    Solver::Enum::default_,
    AdvectionLimiter::Enum::default_,
    Matrix::Enum::default_,
    AdvectionFlux::Enum::upwind,
    DiffusionFlux::Enum::primal > ACAdvDiff;
```

Run-time configuration choices

```
static const std::string gridString[] = { "YaspGrid" , "OneDGrid" , "UGGrid" };
const int gridNr = Fem::Parameter::getEnum( "grid" , probString, 0 );
if( gridNr == 0 )
    return Dune::YaspGrid< dim, Coordinates > ();
else if ( gridNr == 1 )
    return Dune::OneDGrid () ;
else if ( gridNr == 2 )
    return UGGrid< dim > ();
else
{
    abort();
    return 0;
}
```

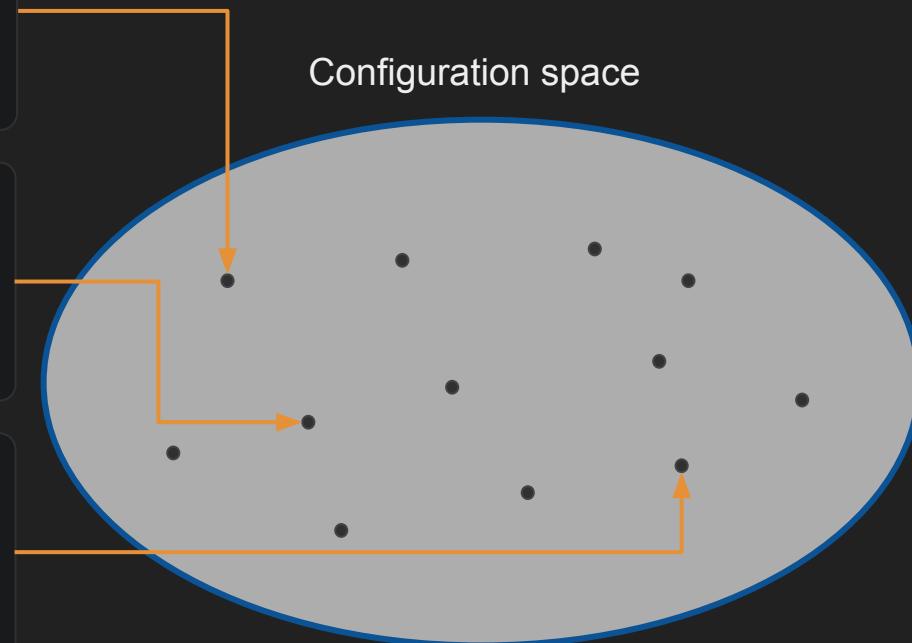
Compile-time configuration choices

```
AlgorithmConfigurator< GridSelectorGridType,  
                      Galerkin::Enum::default_,  
                      Adaptivity::Enum::default_,  
                      DiscreteFunctionSpaces::Enum::gausslobatto,  
                      Solver::Enum::default_,  
                      AdvectionLimiter::Enum::default_,  
                      Matrix::Enum::default_,  
                      AdvectionFlux::Enum::upwind,  
                      DiffusionFlux::Enum::primal > ACAdvDiff;
```

```
AlgorithmConfigurator< GridSelectorGridType,  
                      Galerkin::Enum::default_,  
                      Adaptivity::Enum::default_,  
                      DiscreteFunctionSpaces::Enum::legendre,  
                      Solver::Enum::default_,  
                      AdvectionLimiter::Enum::default_,  
                      Matrix::Enum::default_,  
                      AdvectionFlux::Enum::upwind,  
                      DiffusionFlux::Enum::primal > ACAdvDiff;
```

```
AlgorithmConfigurator< GridSelectorGridType,  
                      Galerkin::Enum::cg,  
                      Adaptivity::Enum::default_,  
                      DiscreteFunctionSpaces::Enum::legendre,  
                      Solver::Enum::default_,  
                      AdvectionLimiter::Enum::default_,  
                      Matrix::Enum::default_,  
                      AdvectionFlux::Enum::upwind,  
                      DiffusionFlux::Enum::primal > ACAdvDiff;
```

Configuration space



Compile-time configuration choices

```
AlgorithmConfigurator< GridSelectorGridType,  
    Galerkin::Enum::default_,  
    Adaptivity::Enum::default_,  
    DiscreteFunctionSpaces::Enum::gausslobatto,  
    Solver::Enum::default_,  
    AdvectionLimiter::Enum::default_,  
    Matrix::Enum::default_,  
    AdvectionFlux::Enum::upwind,  
    DiffusionFlux::Enum::primal > ACAdvDiff;
```

```
AlgorithmConfigurator< GridSelectorGridType,  
    Galerkin::Enum::default_,  
    Adaptivity::Enum::default_,  
    DiscreteFunctionSpaces::Enum::legendre,  
    Solver::Enum::default_,  
    AdvectionLimiter::Enum::default_,  
    Matrix::Enum::default_,
```

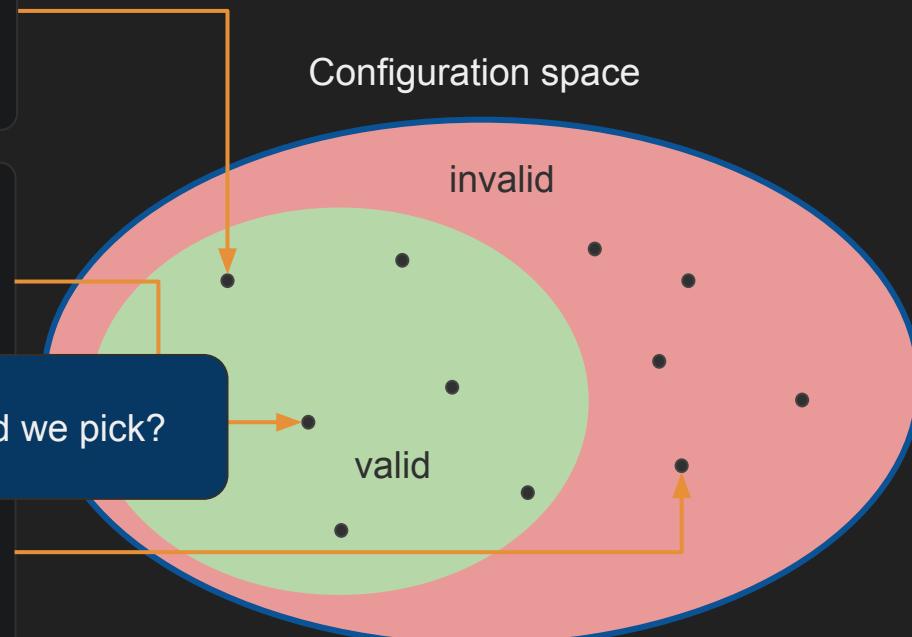
```
AlgorithmConfigurator<  
    Adaptivity::Enum::default_,  
    DiscreteFunctionSpaces::Enum::legendre,  
    Solver::Enum::default_,  
    AdvectionLimiter::Enum::default_,  
    Matrix::Enum::default_,  
    AdvectionFlux::Enum::upwind,  
    DiffusionFlux::Enum::primal > ACAdvDiff;
```

Which valid configuration should we pick?

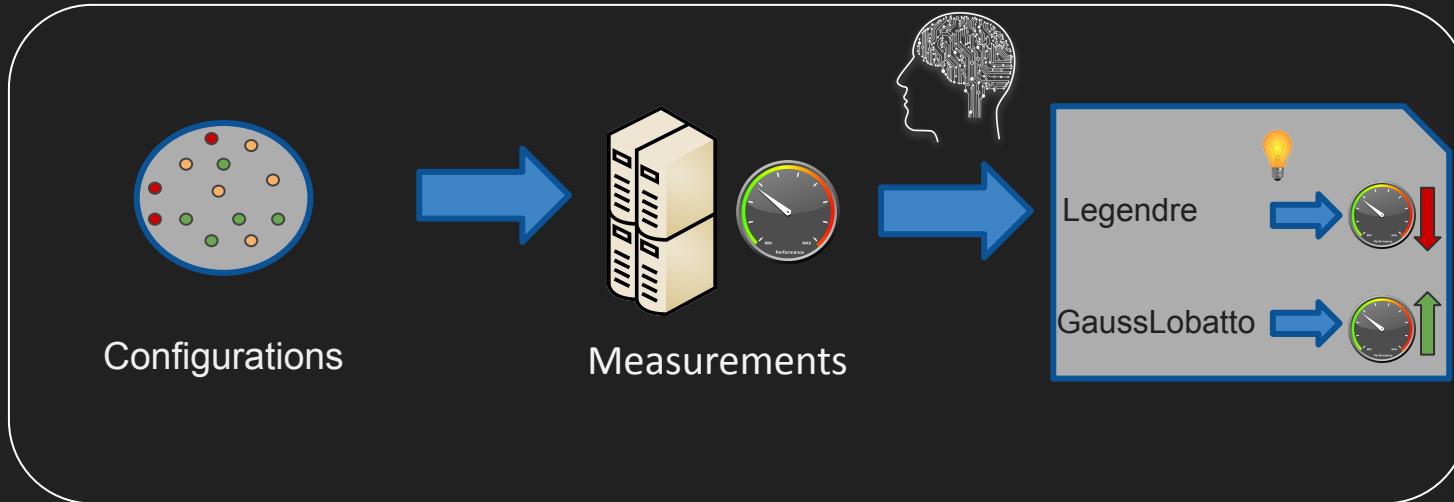
Configuration space

invalid

valid



Black Box



Releases

- ❖ Performance characteristics can change over time

Legendre



GaussLobatto



Legendre



GaussLobatto



Legendre



GaussLobatto



2.7

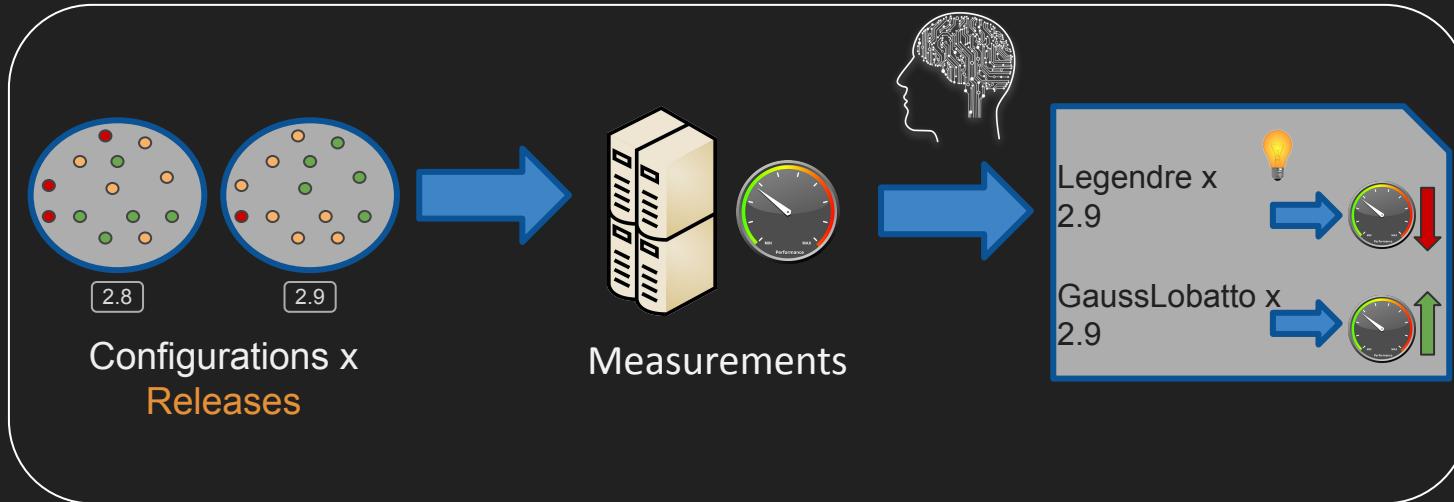
2.8

2.9

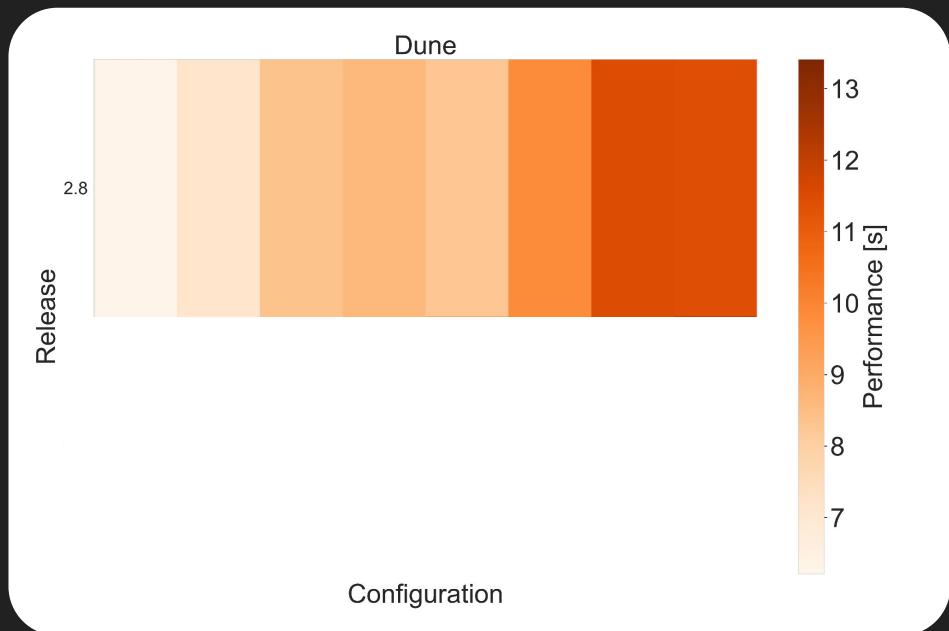
speed up



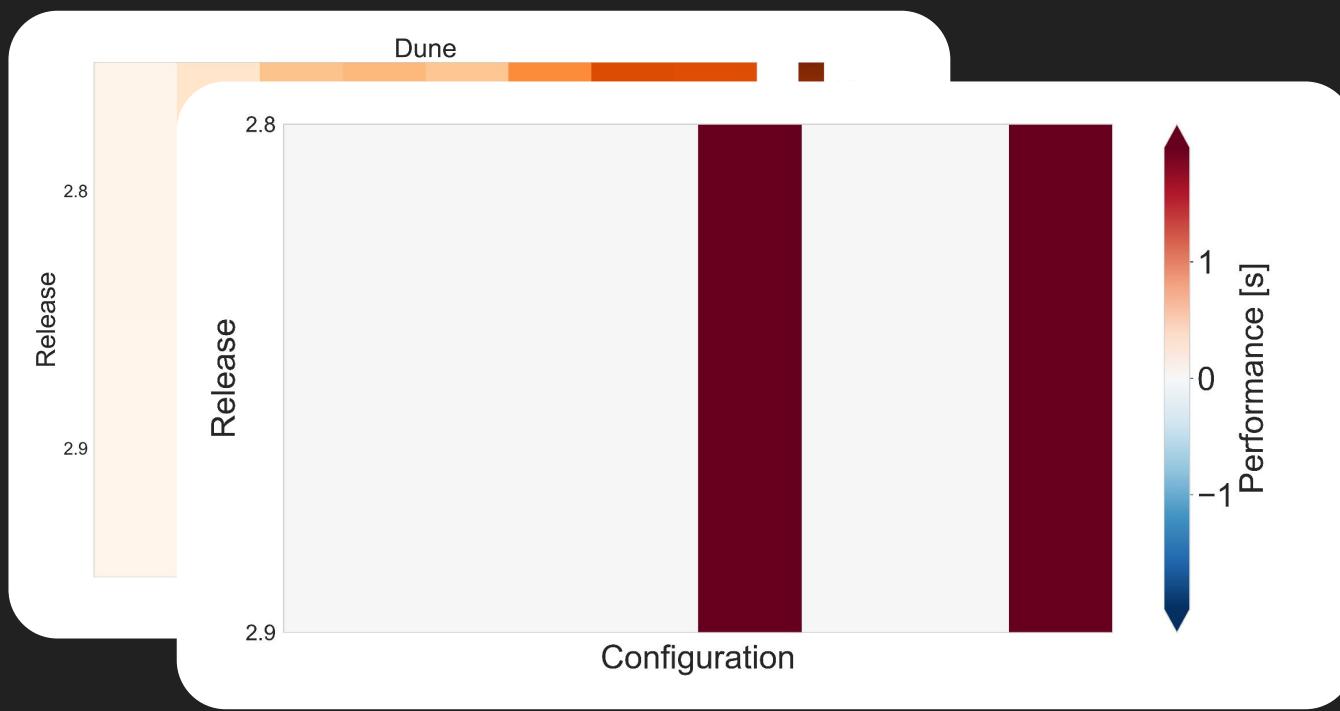
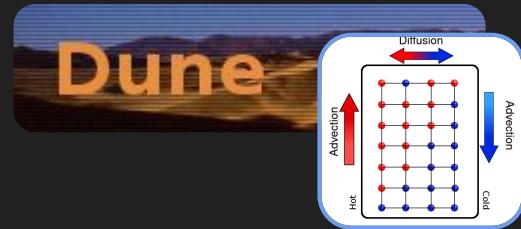
Black Box



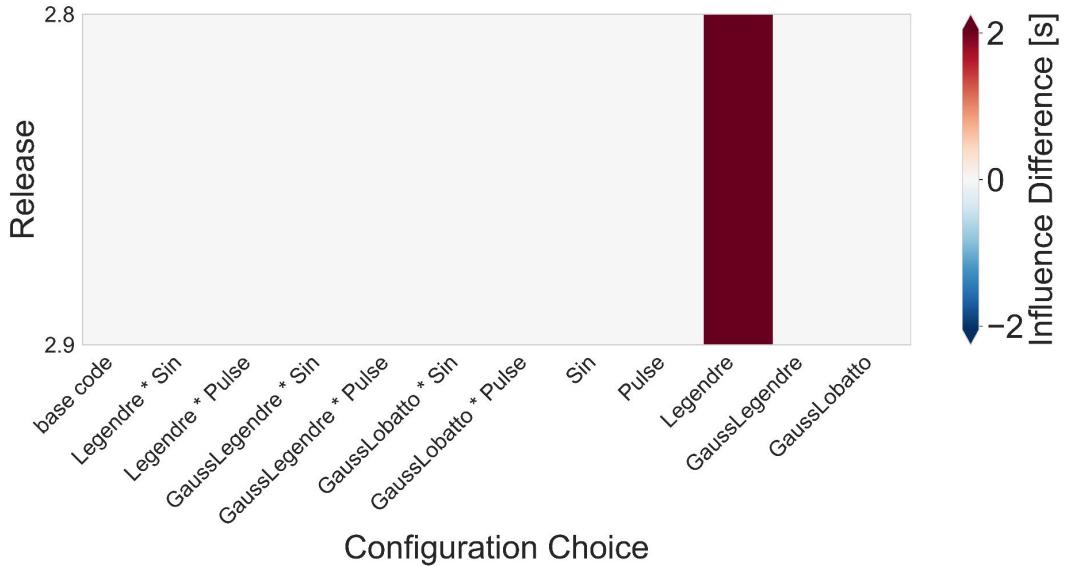
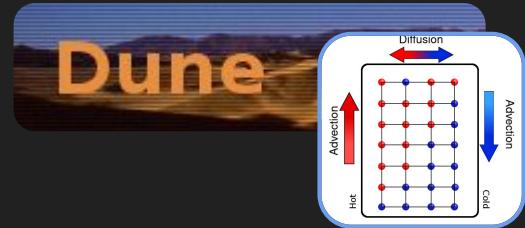
A Historical View on Performance



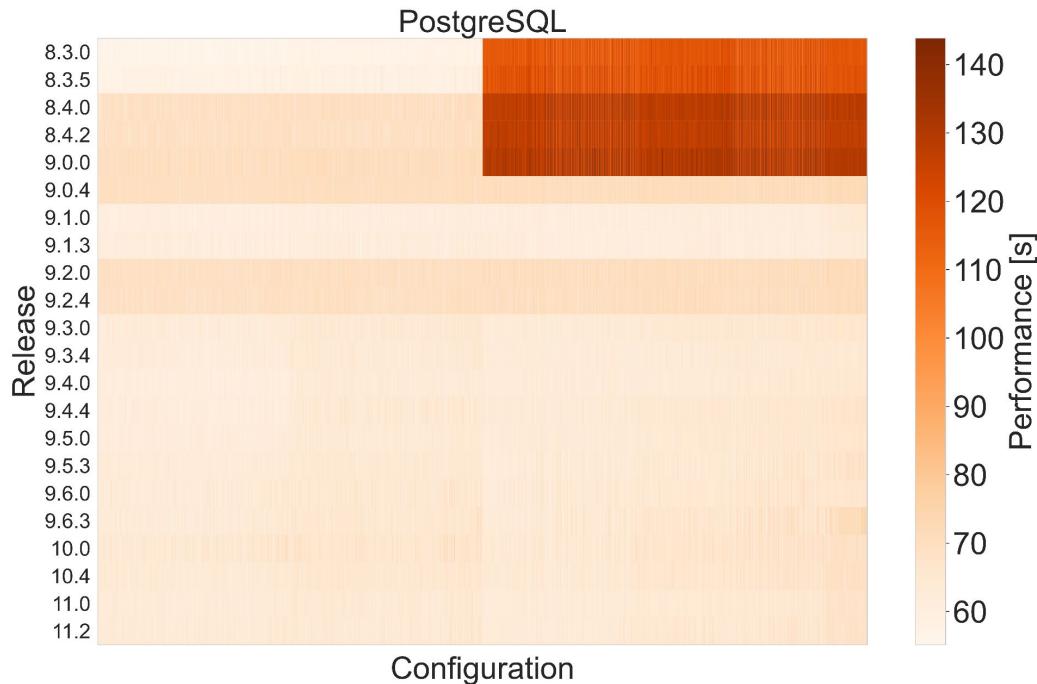
A Historical View on Performance



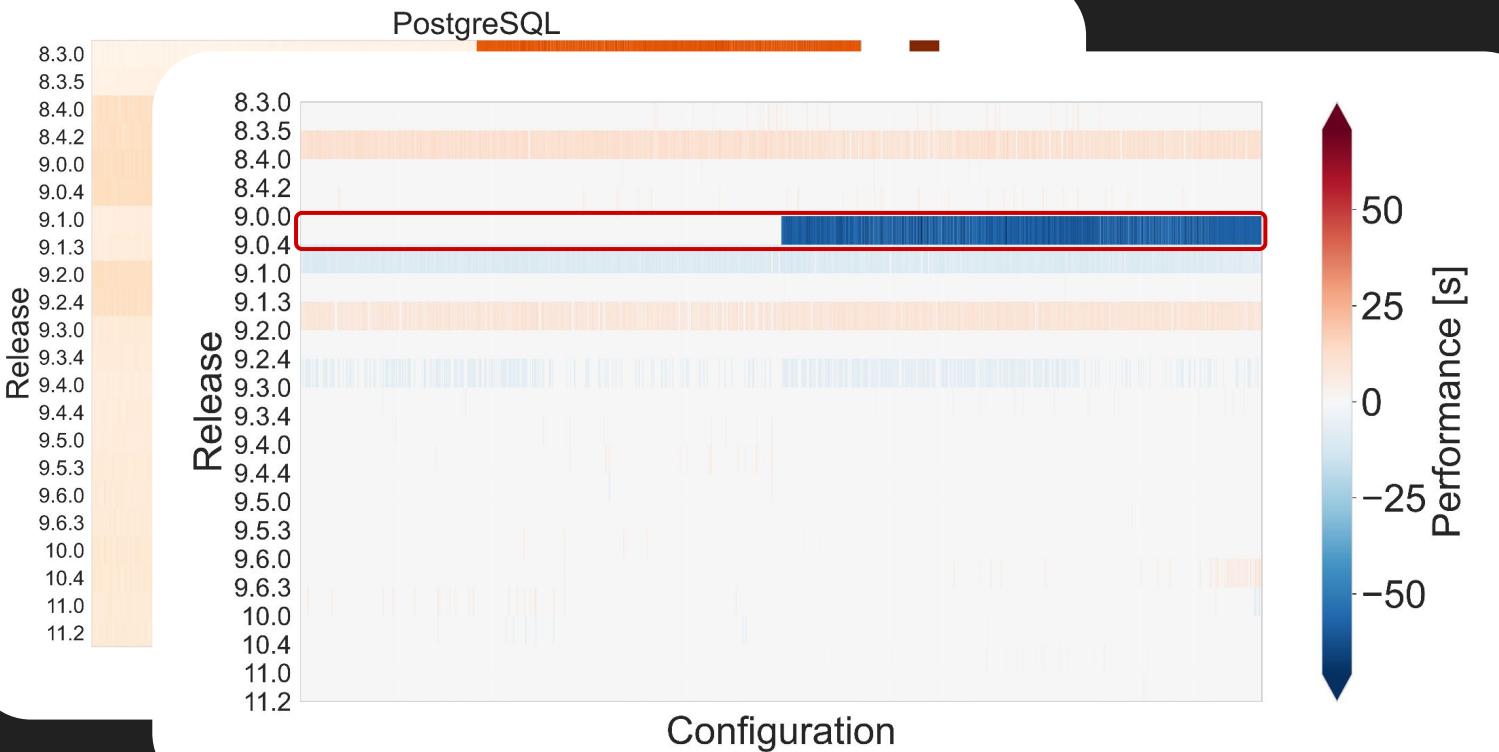
A Historical View on Performance



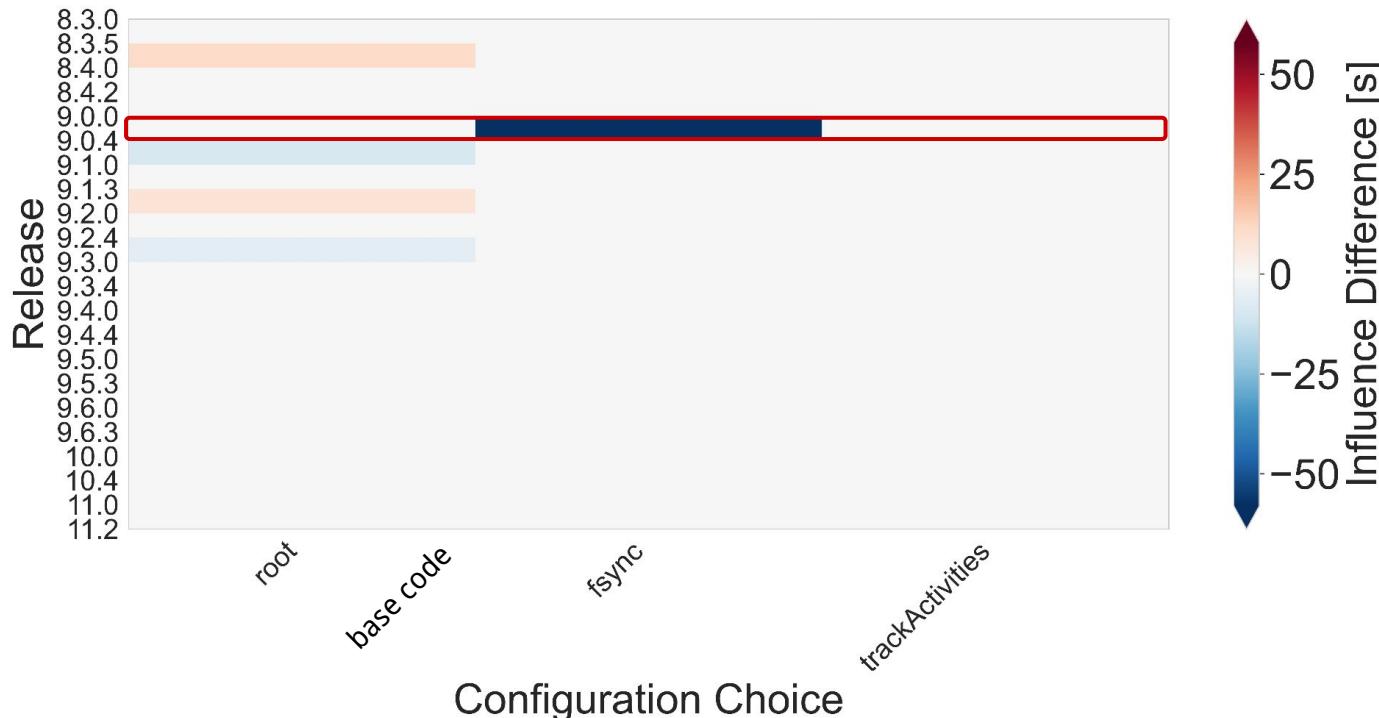
Results on PostgreSQL



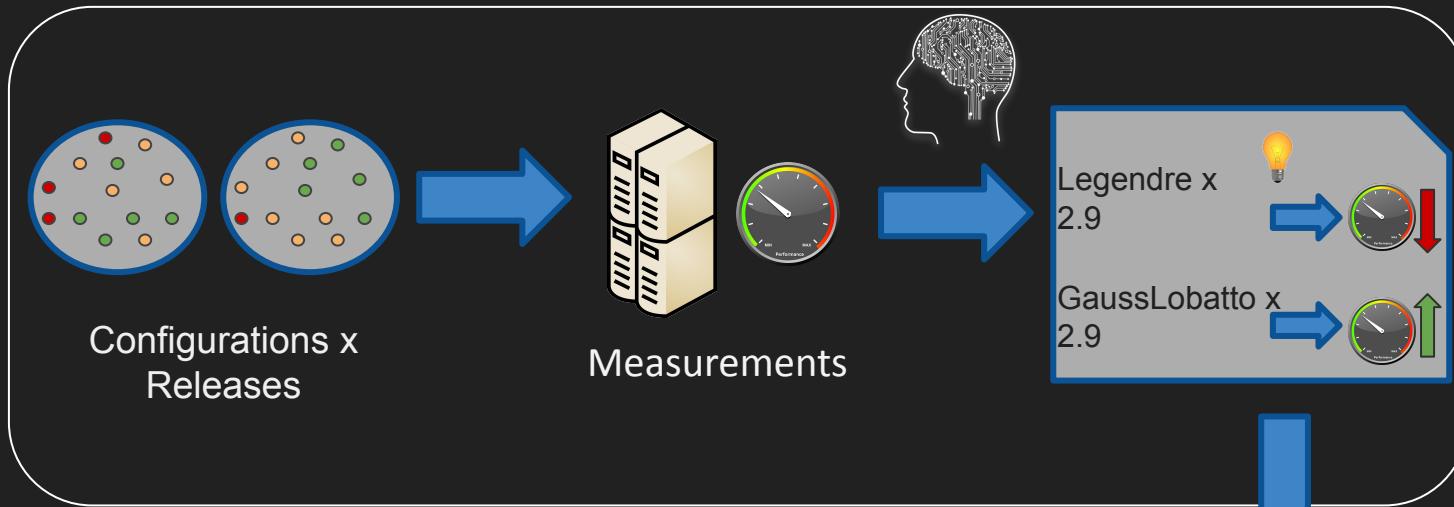
Results on PostgreSQL



Affected Configuration Choices



Black Box



White Box



Search

4280440.8 s +

x

Process 727168

void Dune::Fem::simulate<4, Dune::Fem::AdvectionDiffusionAlgorithmCreator<Dune::ALUGrid<2, 2, (Dune::ALUGridElementType)1, (Dune::ALUGridRefinementType)1, Dune::ALUGridMPIComm>> <(Dune::Fem::AdvectionDiffusionAlgorithmCreator<Dune::ALUGrid<2, 2, (Dune::ALUGridElementType)1, (Dune::ALUGridRefinementType)1, Dune::ALUGridMPIComm>>::SubAdvectionDiffusionAlgorithmCreator<Dune::Fem::AlgorithmC...>

Thread 727168





Search

4280440.8 s +



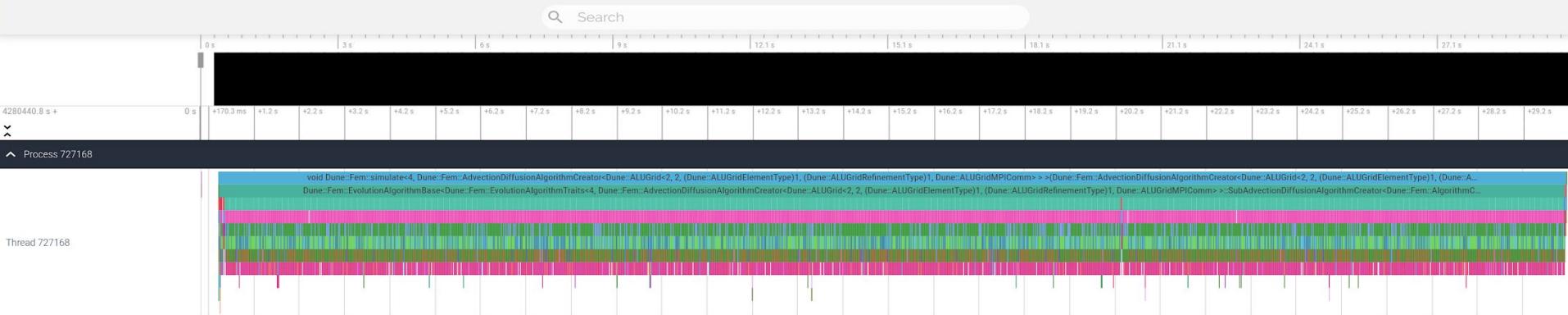
Process 727168

```
void Dune::Fem::simulate<4, Dune::Fem::AdvectionDiffusionAlgorithmCreator<Dune::ALUGrid<2, 2, (Dune::ALUGridElementType)1, (Dune::ALUGridRefinementType)1, Dune::ALUGridMPIComm>>>(Dune::Fem::AdvectionDiffusionAlgorithmCreator<Dune::ALUGrid<2, 2, (Dune::ALUGridElementType)1, (Dune::ALUGridRefinementType)1, Dune::ALUGridMPIComm>>>SubAdvectionDiffusionAlgorithmCreator<Dune::Fem::AlgorithmC...
```

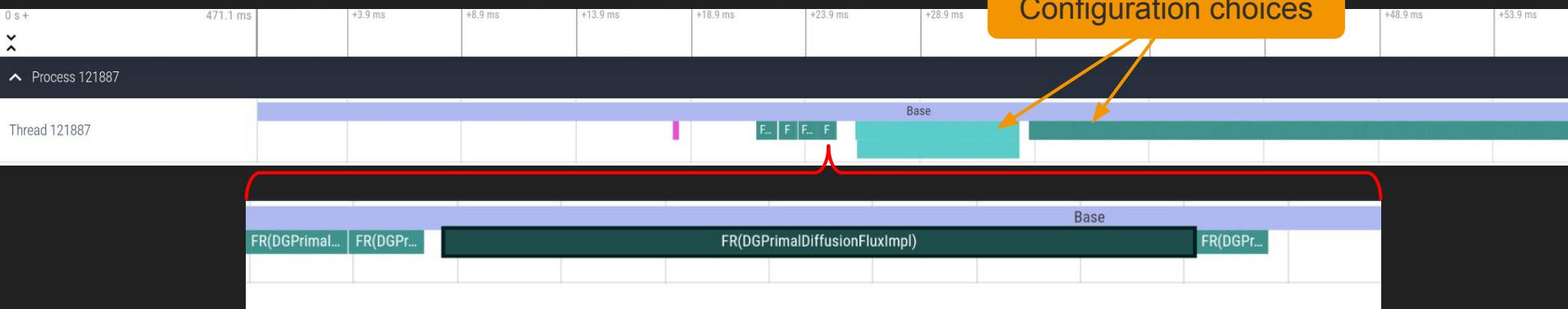
Thread 727168







Configuration choices

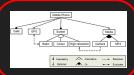




Configurability
Annotations

Dune

```
17 // Dune
18 // dune
19 // boost
20
21 // what's option {
22 //   In <option> {
23 //     member<values>
24 //     member<description, values>
25 //     system<values> = &values;
26 //     system<description> = &description;
27 //     if (values < 0) {
28 //       values = -values;
29 //     }
30 //   }
31 //   else {
32 //     member<values>
33 //     member<description, values>
34 //     system<values> = &values;
35 //     system<description> = &description;
36 //     if (values < 0) {
37 //       values = -values;
38 //     }
39 //   }
40 // }
```



Instrumented Binary



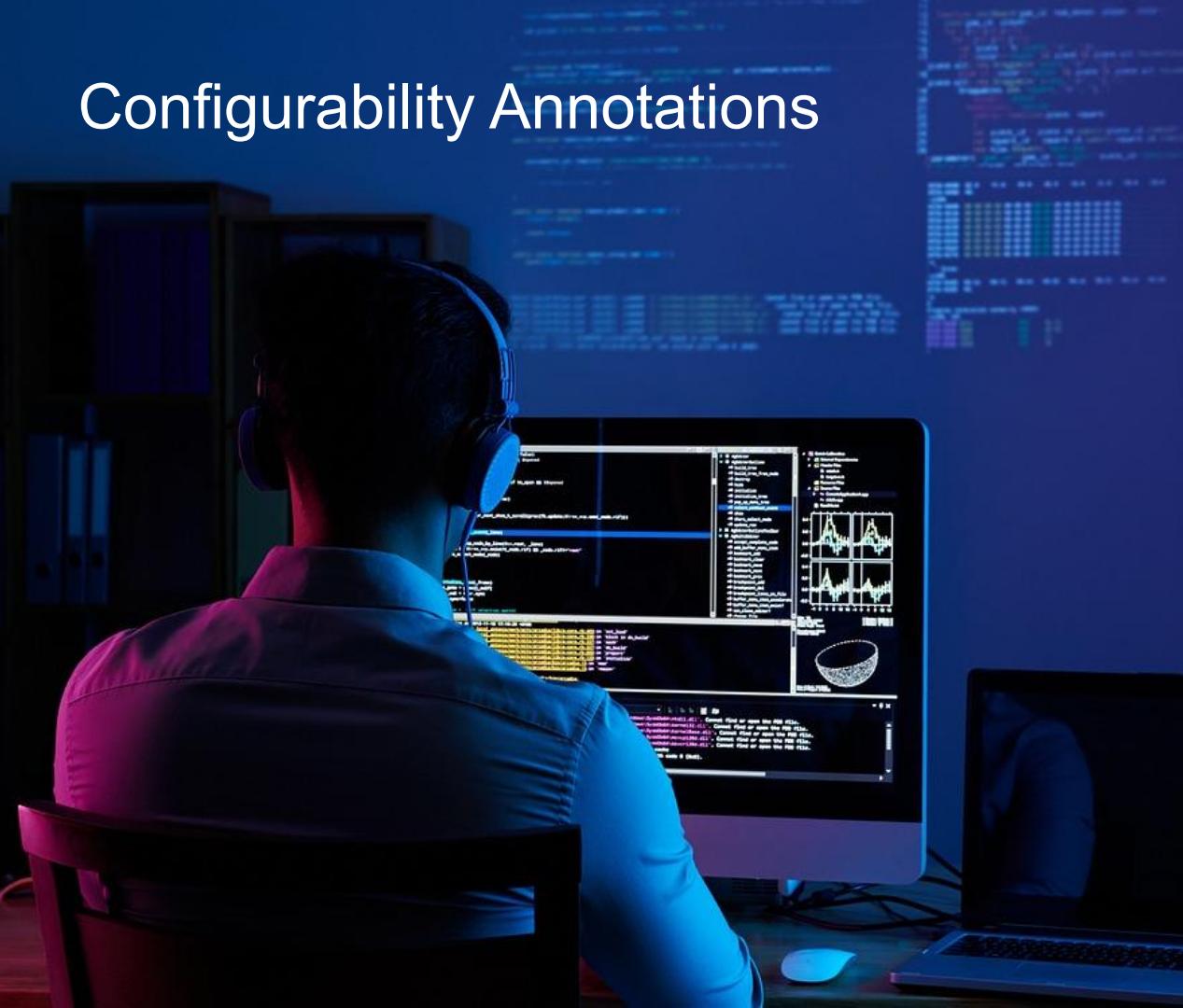
Configuration-Specific Measurements



Configuration-Focused Observability

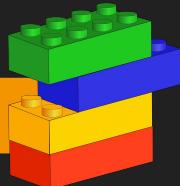


Configurability Annotations



Feature Code

- ❖ class
- ❖ function
- ❖ variable



Configuration-specific class

```
template <class DiscreteFunctionSpace>
class DiscontinuousGalerkinLocalInterpolation {
    typedef DiscontinuousGalerkinLocalInterpolation < DiscreteFunctionSpace > ThisType;
    ...
}
```

Configuration-specific class

```
template <class DiscreteFunctionSpace>
class __attribute__ ((feature("DiscontinuousGalerkinLocalInterpolation ")))
DiscontinuousGalerkinLocalInterpolation { ←
    typedef DiscontinuousGalerkinLocalInterpolation < DiscreteFunctionSpace > ThisType;
...
...
```

Configuration-specific class

```
template <class DiscreteFunctionSpace>
class __attribute__ ((feature("DiscontinuousGalerkinLocalInterpolation ")))
DiscontinuousGalerkinLocalInterpolation {
    typedef DiscontinuousGalerkinLocalInterpolation < DiscreteFunctionSpace > ThisType;
    ...
private:
    template<class QuadImpl, class LocalFunction, class LocalDofVector>
    bool computeInterpolation ( const EntityType& entity,
                               const QuadImpl& quadrature,
                               const LocalFunction &localFunction,
                               LocalDofVector &dofs ) const {
        ...
        return isAffine;
    }
}
```

Configuration-specific code



Configuration-specific function

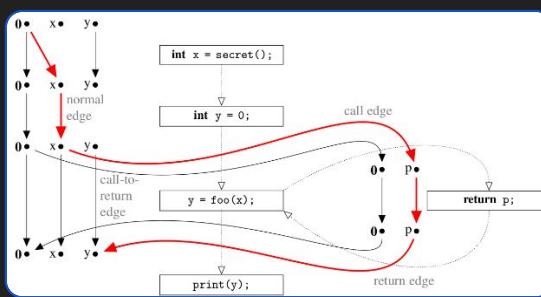
```
bool __attribute__((feature("Configuration-specific Function ")))  
compute (const EntityType& entity) const {  
    ...  
    return isAffine;  
}
```

Configuration-specific code



Configuration variable

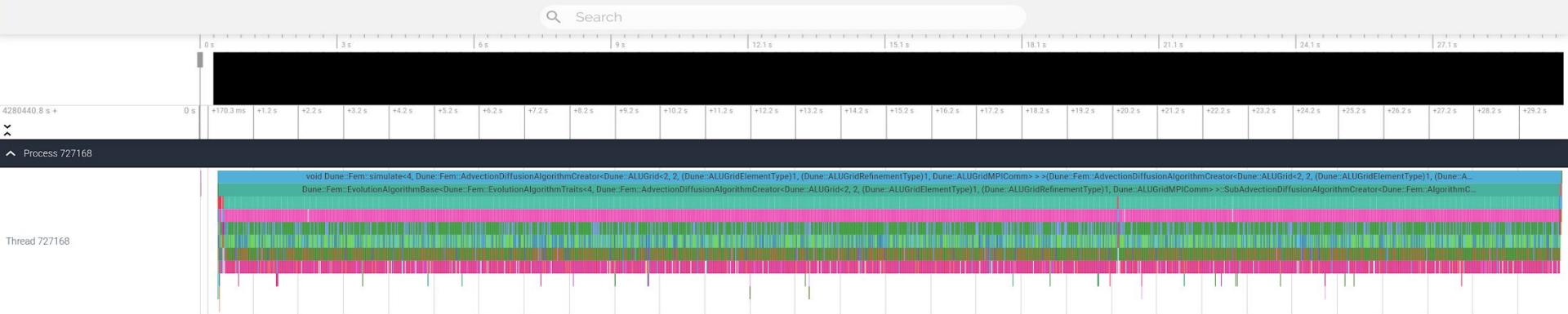
```
// ./myTool --load-time=42  
int __attribute__((feature("Configuration Variable "))) LoadTime = getValueFromUser(argv);  
  
if (LoadTime >= 42) {  
    ...  
}
```

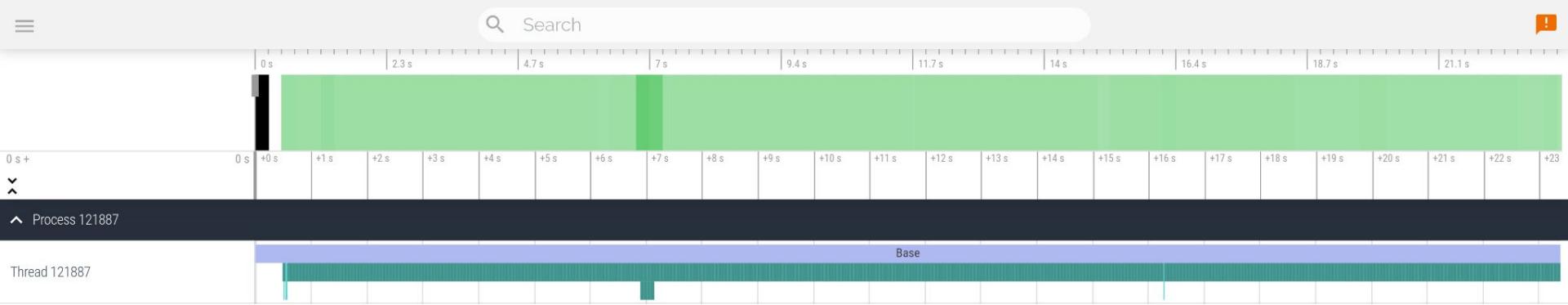


Configuration-specific code



Search

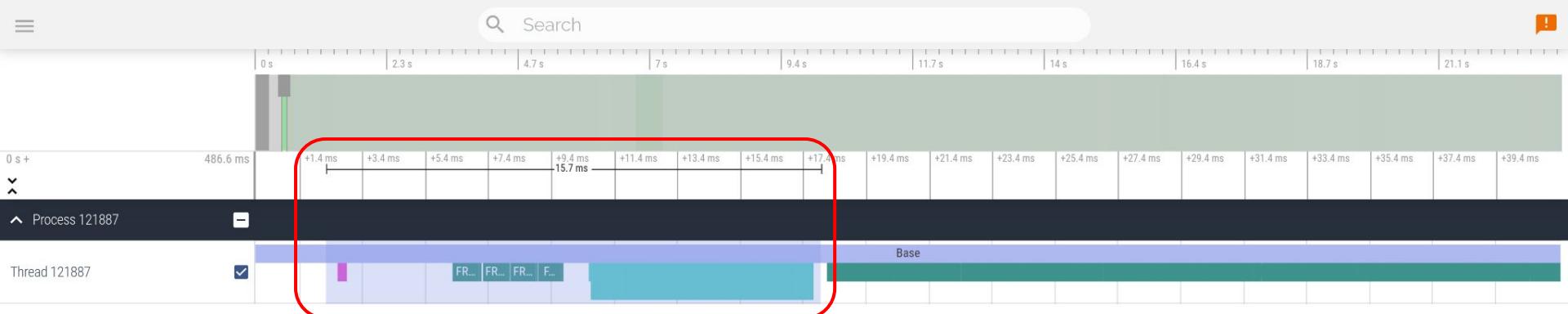




Configuration-Focused Observability

Pivot Table

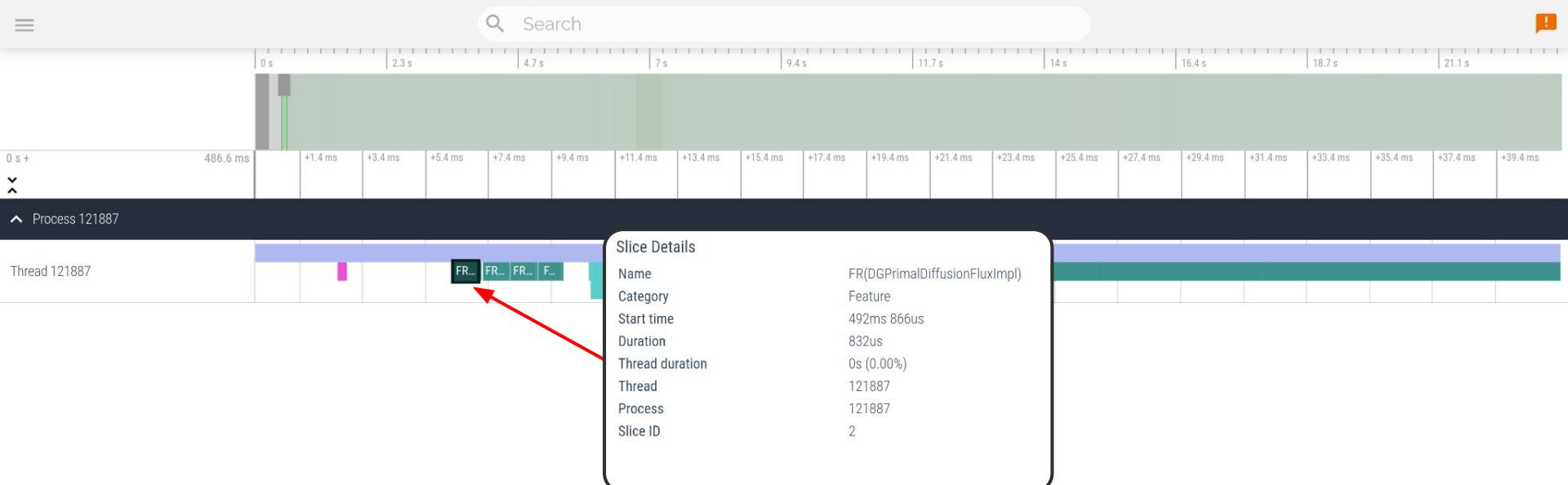
	slice.name...	SUM(slice.dur) ▾	Count...
		Total values:	
Base		32s 129ms 188us	12219271
FR(DGPrimalDiffusionFluxImpl)		23s 395ms 125us	1
FR(DiscontinuousGalerkinLocalInterpolation)		8s 704ms 889us	12213123
FR(ALUGrid)		28ms 890us	6146
		284us	1



Configuration-Focused Observability

Pivot Table

	slice.name...	SUM(slice.dur) ▾	Count...	≡
		Total values: 23s 409ms 675us	2060	↑
Base		23s 395ms 125us	1	↗
FR(DiscontinuousGalerkinLocalInterpolation)		10ms 853us	2050	↗
FR(DGPrimalDiffusionFluxImpl)		3ms 413us	8	↗
FR(ALUGrid)		284us	1	↗



Configuration-Focused Observability

Pivot Table

	slice.name...	SUM(slice.dur) ▾	Count...	≡
		Total values: 23s 409ms 675us	2060	↑
Base		23s 395ms 125us	1	↗
FR(DiscontinuousGalerkinLocalInterpolation)		10ms 853us	2050	↗
FR(DGPrimalDiffusionFluxImpl)		3ms 413us	8	↗
FR(ALUGrid)		284us	1	↗

What Measurement Tools Are Supported?



Adapter Strategies



tool provides custom instrumentation code
→ initialize, finalize, start, end

→ By injection



configuration-specific mapping information is persisted
→ e.g., USDT_probes, custom binary section

→ By interpretation

A black and white photograph of a man from behind, looking through a telescope. He is wearing a dark long-sleeved shirt. The telescope's eyepiece is held close to his eye. The background shows a bright, cloudy sky, likely during sunset or sunrise, with dark silhouettes of trees in the foreground.

**WHAT ARE WE
LOOKING FOR?**

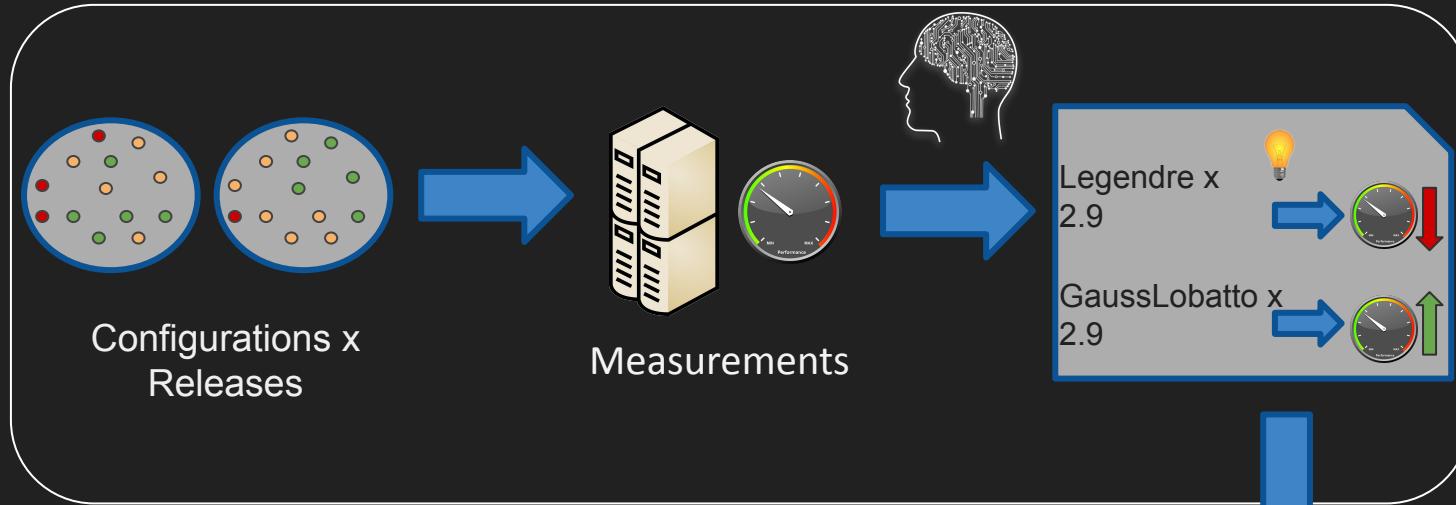
Looking for Real-World Case Studies with



- ❖ configurable code (preferred class/function based)
- ❖ coarse/fine grained abstractions
- ❖ some with clear performance distinctions
- ❖ benchmarks/example workloads
- ❖ contact to domain experts

Overview

Black Box



White Box



VaRA