< Multigrid > < Factorization < ValueType, IndexType > > gko::EnableLinOp< Factorization < ValueType, IndexType > > < Isai< IsaiType, ValueType, IndexType > > gko::EnableLinOp< Isai < IsaiType, ValueType,</pre> IndexType > > < SparsityCsr< ValueType, gko::EnableLinOp< Sparsity Csr< ValueType, IndexType > > IndexType > > < Diagonal < ValueType > > gko::EnableLinOp< Diagonal < ValueType > > < Pgm< ValueType, IndexType > > gko::EnableLinOp< Pgm < ValueType, IndexType > > < Dense< ValueType > > gko::EnableLinOp< Dense < ValueTypė > > < UpperTrs< ValueType, IndexType > > gko::EnableLinOp< UpperTrs < ValueType, IndexType > > < Hybrid< ValueType, IndexType > > gko::EnableLinOp< Hybrid < ValueType, IndexType > > 11/1/1// gko::EnableLinOp< Identity < ValueType > > < Identity< ValueType > > < Fft3 > gko::EnableLinOp< Fft3 > < Composition< ValueType > > gko::EnableLinOp < Composition < ValueType > > < RowGatherer< IndexType > > gko::EnableLinOp < RowGatherer < IndexType > > < Fft2 > 11111 gko::EnableLinOp< Fft2 > < FixedCoarsening< gko::EnableLinOp< FixedCoarsening < ValueType, IndexType > > ValueType, IndexType > > < Fbcsr< ValueType, gko::EnableLinOp< Fbcsr < ValueType, IndexType > > IndexType > > gko::EnableAbstractPolymorphic < ConcreteLinOp, Polymorphic gko::EnableLinOp < Bicgstab < Bicgstab< ValueType > > Object < ConcreteLinOp, Polymorphic Base > < ValueType > > Base > gko::EnableAbstractPolymorphic PolymorphicBase < < ConcreteObject, Polymorphic Object < AbstractObject, Polymorphic < ConcreteLinOp, Polymorphic Base > Base > < LowerTrs< ValueType, gko::EnablePolymorphicObject \_Base > \_ \_ \_ \_ gko::EnablePolymorphicObject < ConcreteLinOp, PolymorphicBase > gko::EnableLinOp< LowerTrs gko::EnableAbstractPolymorphic IndexType > > < ConcreteObject, PolymorphicBase > < ValueType, IndexType > > Object < ConcreteObject, Polymorphic gko::EnableLinOp< Concrete LinOp, PolymorphicBase > Base > < Combination< ValueType > > gko::EnableLinOp< Combination < ValueType > > gko::ConvertibleTo gko::EnablePolymorphicAssignment < ConcreteLinOp > < Multigrid > < ConcreteLinOp > < ConcreteLinOp > < ConcreteLinOp > gko::ConvertibleTo < ResultType > gko::EnableLinOp< Multigrid > gko::EnablePolymorphicAssignment < Gmres< ValueType > > MI / / < ConcreteType, ResultType > MIII. MI 1 1 / gko::EnableLinOp< Gmres MI 1/ 1 < ValueType > > < CbGmres< ValueType > > MMI 177 NATE 171 MULL 1 / gko::EnableLinOp< CbGmres MULL 1 1 < ValueType > > MINIT I I < Csr< ValueType, IndexType > > Million 1 / Multiply mm to to the gko::EnableLinOp< Csr HILL I I < ValueType, IndexType > > THILL I - I < Ir< ValueType > > THEFT ! gko::EnableLinOp< Ir THINK ! ! < ValueType > > DEBIN 1 THIIII 1 < Coo< ValueType, IndexType > > THUIL I gko::EnableLinOp< Coo < ValueType, IndexType > > MINIMA A THEFT ! THULLY / THEFT ! < Fcg< ValueType > > gko::EnableLinOp< Fcg < ValueType > > - MILLION / 111111111 1111111111 11111111111 < Fft > THULLY gko::EnableLinOp< Fft > THULL ! 1111111 / < Direct< ValueType, 11/1/1/1/1 IndexType > > gko::EnableLinOp< Direct 11/1/1/1/1 < ValueType, IndexType > > 1111111 -11 H H / < Idr< ValueType > > 1111111 gko::EnableLinOp< Idr 1111111<sup>1</sup>, < ValueType > > 11111111 < Cgs< ValueType > > 11/1/1/1/1/1 gko::EnableLinOp< Cgs  $= \Pi \cdot I \cdot I \cdot I \cdot I \cdot I$ < ValueType > > H + I + I + IM(I) = I + I< Ell< ValueType, IndexType > > gko::EnableLinOp< Ell -H + I + I + I< ValueType, IndexType > > 11/1/1/ < Ilu< LSolverType, USolverType, ReverseApply, IndexType > > H = I + I = I-H + I + Igko::EnableLinOp< llu < LSolverType, USolverType, ReverseApply, IndexType > > -H - I - I - IH = I = IH + I + I< lc< LSolverType, H = I - IIndexType > > 11 1 1 gko::EnableLinOp< Ic < LSolverType, IndexType > > I = I - IH = I = I-H - I - I< Permutation< IndexType > > gko::EnableLinOp < Permutation < IndexType > > < ScaledPermutation < ValueType, IndexType > > gko::EnableLinOp< Scaled Permutation < ValueType, IndexType > >

Dense< value\_type >

gko::solver::Multigrid

< Jacobi< ValueType, IndexType > >

< Gcr< ValueType > >

gko::EnableLinOp< Jacobi < ValueType, IndexType > >

> gko::EnableLinOp< Gcr < ValueType > >