spaceTimeWindow

1.0

Generated by Doxygen 1.8.14

Contents

1 spaceTimeWindow Library **2**

2 Namespace Index **12**
2.1 Namespace List . . . 12

3 Hierarchical Index **13**
3.1 Class Hierarchy . . . 13

4 Class Index **13**
4.1 Class List . . . 13

5 File Index **13**
5.1 File List . . . 13

6 Namespace Documentation **14**
6.1 Foam Namespace Reference . . . 14
6.1.1 Function Documentation . . . 14
6.2 Foam::functionObjects Namespace Reference . . . 14
6.2.1 Function Documentation . . . 15

7 Class Documentation **15**
7.1 Foam::deltaVarintCodec Class Reference . . . 15
7.1.1 Detailed Description . . . 15
7.1.2 Member Function Documentation . . . 16
7.2 Foam::sodiumCrypto Class Reference . . . 18
7.2.1 Detailed Description . . . 18
7.2.2 Member Function Documentation . . . 19
7.2.3 Member Data Documentation . . . 21
7.3 Foam::functionObjects::spaceTimeWindowExtract Class Reference . . . 22
7.3.1 Detailed Description . . . 23
7.3.2 Constructor & Destructor Documentation . . . 24
7.3.3 Member Function Documentation . . . 24
7.3.4 Member Data Documentation . . . 28
7.4 Foam::spaceTimeWindowFvPatchField< Type > Class Template Reference . . . 32
7.4.1 Detailed Description . . . 33
7.4.2 Constructor & Destructor Documentation . . . 33
7.4.3 Member Function Documentation . . . 34

8 File Documentation **36**
8.1 deltaVarintCodec.C File Reference . . . 36
8.2 deltaVarintCodec.H File Reference . . . 36
8.3 README.md File Reference . . . 36
8.4 sodiumCrypto.C File Reference . . . 36
8.5 sodiumCrypto.H File Reference . . . 36
8.6 spaceTimeWindowExtract.C File Reference . . . 37
8.6.1 Function Documentation . . . 38
8.7 spaceTimeWindowExtract.H File Reference . . . 39
8.8 spaceTimeWindowFvPatchField.C File Reference . . . 39
8.9 spaceTimeWindowFvPatchField.H File Reference . . . 39
8.10 spaceTimeWindowFvPatchFields.C File Reference . . . 40
8.11 spaceTimeWindowFvPatchFields.H File Reference . . . 40
8.12 spaceTimeWindowInitCase.C File Reference . . . 40
8.12.1 Function Documentation . . . 41
8.13 spaceTimeWindowKeygen.C File Reference . . . 42
8.13.1 Function Documentation . . . 43

Index **45**

1 spaceTimeWindow Library

A library for space-time window extraction and reconstruction of LES simulations, developed for OpenFOAM.

Tested with: OpenFOAM v2512 (openfoam.com)

> **Trademark Notice:** This offering is not approved or endorsed by OpenCFD Limited, producer and distributor of the OpenFOAM software via www.openfoam.com, and owner of the OPENFOAM and OpenCFD trade marks. OPENFOAM is a registered trade mark of OpenCFD Limited, producer and distributor of the OpenFOAM software via www.openfoam.com.

Overview

This library provides three components for extracting a spatial subset from a full LES simulation and reconstructing the flow within that subset using pre-computed boundary conditions:

1. **spaceTimeWindowExtract** - Function object to extract boundary data during the original simulation
2. **spaceTimeWindowInitCase** - Utility to initialize a reconstruction case from extracted data
3. **spaceTimeWindow** - Boundary condition to apply extracted data during reconstruction

Workflow

The workflow is strictly sequential:

Serial Execution

1. **Extraction phase**: Run the original simulation with `spaceTimeWindowExtract` function object
2. **Case setup**: Run `spaceTimeWindowInitCase` to configure the reconstruction case
3. **Reconstruction phase**: Run the solver directly - everything is pre-configured

Parallel Execution

1. **Extraction phase**: Run the original simulation in parallel with `spaceTimeWindowExtract` function object
   - The extractor automatically forces a field write at the extraction start time
   - Boundary data is gathered from all processors and written by master
2. **Reconstruct fields**: Run `reconstructPar -time <extractionStartTime>` to reconstruct the initial time
3. **Case setup**: Run `spaceTimeWindowInitCase -sourceCase <path>` to create subset mesh and initial fields
4. **Reconstruction phase**: Run the solver on the subset case

```
# Example parallel workflow
cd source-case
mpirun -np 4 pimpleFoam -parallel    # Extraction happens automatically

# Reconstruct the extraction start time
reconstructPar -time 0.0001          # Use actual extraction start time

# Initialize the subset case
cd ../subset-case
spaceTimeWindowInitCase -sourceCase ../source-case

# Run reconstruction (can be serial or parallel)
pimpleFoam
```

Components

spaceTimeWindowExtract (Function Object)

Extracts face-interpolated boundary values at every timestep during the original simulation.

```
functions
{
    extractSubset
    {
        type            spaceTimeWindowExtract;
        libs            (spaceTimeWindow);

        // Define extraction region with bounding box (min max points)
        // Box must be fully internal to domain (no intersection with boundaries)
        box             ((0.05 -0.25 0.01) (0.90 0.25 0.38));

        outputDir       "../subset-case";        // Output case directory
        fields          (U p nut);               // Fields to extract

        // Write format for boundary data (optional)
        // Options: ascii, binary, deltaVarint
        // Default: ascii
        writeFormat     deltaVarint;

        // Precision for deltaVarint (optional, default: 6)
        deltaVarintPrecision  6;

        // Compression for ascii/binary (optional, ignored for deltaVarint)
        // writeCompression on;

        writeControl    timeStep;
        writeInterval   1;
    }
}
```

Parameters:

Parameter	Description	Required	Default
box	Bounding box as ((minX minY minZ) (maxX maxY maxZ))	yes	-
outputDir	Output case directory for extracted data	yes	-
fields	List of fields to extract	yes	-
writeFormat	Format for boundary data: `ascii`, `binary`, or `deltaVarint`	no	ascii
deltaVarintPrecision	Decimal digits for delta-varint quantization	no	6
writeCompression	Gzip compression for ascii/binary (ignored for deltaVarint)	no	off

Field Selection:

The `fields` parameter explicitly lists which fields to extract. Fields are **not** auto-detected from the turbulence model. Common configurations:

Turbulence Model	Recommended fields
LES Smagorinsky	`(U p nut)`
LES dynamicKEqn	`(U p nut k)`
RANS k-epsilon	`(U p nut k epsilon)`
RANS k-omega SST	`(U p nut k omega)`

If a requested field doesn't exist in the simulation, extraction will fail with a "field not found" error.

Output structure:

```
outputDir/
    constant/
        polyMesh/               # Subset mesh (only oldInternalFaces patch)
        boundaryData/
            oldInternalFaces/
                points              # Face centres (reference only)
                extractionMetadata  # Settings and timestep list
                <time>/
                    U               # Face-interpolated velocity (or U.dvz)
                    p               # Face-interpolated pressure (or p.dvz)
                    nut             # Face-interpolated turbulence viscosity (or nut.dvz)
    <startTime>/                # Initial subset fields (always ASCII)
        U
        p
        nut
```

Note: With `writeFormat deltaVarint`, boundary data files have `.dvz` extension (e.g., `U.dvz`). Initial fields remain ASCII regardless of writeFormat setting.

Notes:

- Box must be **fully internal** to the domain (no intersection with external boundaries)
- The mesh is written with only `oldInternalFaces` patch (type `patch`)
- Boundary data is written at every timestep (in `execute()`), not just at write intervals
- The `extractionMetadata` file includes the list of all extracted timesteps (exact directory names)
- **Supports both serial and parallel execution**
- Face values are computed using linear interpolation: `U_face = w * U_inside + (1-w) * U_outside`

Parallel Execution:

- The extraction box can span multiple processor domains
- Boundary data from all processors is gathered to the master and written as a single file
- Processor boundary faces (where the extraction boundary crosses processor boundaries) are handled automatically
- Field interpolation correctly exchanges values between processors for faces on processor boundaries
- **Automatic field write**: At extraction start time, the extractor forces the solver to write all fields to disk
- **Mesh handling**: In parallel mode, only extraction box parameters are written (not the mesh)
- `spaceTimeWindowInitCase` creates the subset mesh from the reconstructed source case, ensuring correct cell ordering

Parallel Extraction Output:

```
outputDir/
    constant/
        polyMesh/
            extractionBox       # Extraction parameters for mesh creation
        boundaryData/
            oldInternalFaces/
                points              # Face centres (gathered from all processors)
                extractionMetadata  # Settings, timestep list, nProcs info
                <time>/
                    U               # Gathered face-interpolated fields (or U.dvz)
                    p               # (or p.dvz with deltaVarint)
                    nut             # (or nut.dvz)
```

Important: After parallel extraction, you must run `reconstructPar -time <startTime>` before `spaceTimeWindowInitCase`.

spaceTimeWindowInitCase (Utility)

Initializes a fully configured reconstruction case from the extracted data.

```
# Run from the extraction output directory
spaceTimeWindowInitCase -sourceCase ../ufr2-02

# Or specify the extract directory explicitly
spaceTimeWindowInitCase -sourceCase ../ufr2-02 -extractDir ./subset-case
```

Options:

Option	Description	Required
-sourceCase	Source case directory (where extraction ran)	yes
-extractDir	Directory with extracted data (default: cwd)	no
-overwrite	Overwrite existing files	no

What it creates:

1. `system/controlDict` - With matching solver, deltaT, adjustTimeStep from extraction
   - **Note:** `endTime` is set to the **second-to-last** available timestep, because the `spaceTimeWindow` BC reads ahead to the next timestep
2. `system/fvSchemes`, `system/fvSolution` - Copied from source case
3. `constant/` files - All physics properties copied (mandatory for fidelity):
   - `turbulenceProperties`, `transportProperties`
   - `momentumTransport`, `thermophysicalProperties`
   - `LESProperties`, `RASProperties`, `g`
4. Initial field files with `spaceTimeWindow` BC on `oldInternalFaces`
5. Turbulence fields (nut, k, epsilon, omega, etc.) with appropriate wall functions

Parallel case handling:

When the extraction was done in parallel (detected by `extractionBox` file):

- Creates subset mesh from the reconstructed source case using the extraction bounding box
- Extracts initial fields from the source case at the extraction start time
- Requires that `reconstructPar -time <startTime>` was run on the source case first

After running:

```
cd subset-case
pimpleFoam    # Run directly - everything is configured
```

spaceTimeWindow (Boundary Condition)

Reads pre-computed face values from boundaryData and applies them as boundary conditions.

```
oldInternalFaces
{
    type            spaceTimeWindow;
    dataDir         "constant/boundaryData";    // Path to boundaryData
    fixesValue      true;                       // See mass conservation section
    value           uniform (0 0 0);
}
```

Properties:

Property	Description	Required	Default
dataDir	Path to boundaryData directory	no	constant/boundaryData
fieldTableName	Name of field file (if different from field)	no	field name
setAverage	Adjust mapped field to match average	no	false
offset	Offset value added to field	no	Zero
fixesValue	Report to adjustPhi that values are fixed	no	true

Notes:

- Does NO spatial interpolation (values are pre-computed for exact face positions)
- Does NO temporal interpolation - requires exact timestep matching with source simulation
- Will error if simulation time doesn't match an available timestep
- Metadata validation enforces identical `deltaT` to ensure timestep alignment

Mass Conservation and `fixesValue`

The `fixesValue` option controls how `adjustPhi()` handles flux correction on the `oldInternalFaces` patch.

fixesValue = true (default)

- `adjustPhi()` excludes this patch from flux correction
- Preserves exact boundary values from original simulation
- Any mass imbalance is corrected on other adjustable patches (typically outlets)
- Use when: You want to preserve the exact extracted boundary values

fixesValue = false

- `adjustPhi()` includes this patch in flux correction
- Allows small modifications to the flux to ensure mass conservation
- The correction is distributed across all faces of this patch
- Use when: You want `adjustPhi()` to heal/repair mass imbalance caused by face interpolation

Recommendation: For LES reconstruction where accurate inflow physics matter, `fixesValue = false` may be preferable as it distributes the mass correction uniformly across the `oldInternalFaces` rather than dumping it all on the outlet.

Boundary Data Compression

The `writeFormat` parameter controls how boundary data files are written. For long LES simulations with many timesteps, storage can become significant.

Format Comparison

Format	Extension	Typical Size	Notes
`ascii`	(none)	100% baseline	Human-readable, default
`binary`	(none)	~50%	OpenFOAM native binary
`ascii` + gzip	.gz	~10%	`writeCompression on`
`binary` + gzip	.gz	~8%	`writeFormat binary` + `writeCompression on`
`deltaVarint`	.dvz	~2.7%	Best compression, recommended

ASCII and Binary Formats

Standard OpenFOAM formats with optional gzip compression:

```
// ASCII (default, human-readable)
writeFormat     ascii;

// Binary (smaller, faster I/O)
writeFormat     binary;

// Either format with gzip compression
writeCompression on;
```

Delta-Varint Codec (`writeFormat deltaVarint`)

A specialized codec optimized for time-series CFD data that achieves ~97% storage reduction:

How it works:

1. **Component-major ordering**: Stores all Ux values, then Uy, then Uz (groups similar values)
2. **Delta encoding**: Stores differences between consecutive values (small deltas)
3. **Quantization**: Rounds deltas to configurable precision (default: 6 decimal digits)
4. **Varint encoding**: Variable-length integer encoding (1-9 bytes based on magnitude)
5. **Zigzag encoding**: Efficient encoding of signed integers

File format:

- Magic number: `DVZ1` (0x315A5644)
- Header: element count, component count, precision
- First value: stored as raw 8-byte double (exact)
- Subsequent values: delta-encoded varints

Precision:

- `deltaVarintPrecision 6` means deltas quantized to `round(delta * 10^6)`
- Precision loss is ~1e-6 relative to consecutive value differences
- For typical CFD data, this is well below solver tolerance

What uses deltaVarint:

- Boundary data files only (`constant/boundaryData/.../U.dvz`, `p.dvz`)

Important: When using `writeFormat deltaVarint`, all other output is written as **uncompressed ASCII**:

- Initial fields (`<startTime>/U`, `p`) - ASCII
- Subset mesh (`constant/polyMesh/`) - ASCII
- Metadata (`extractionMetadata`) - ASCII

The `writeCompression` setting is **ignored** when using deltaVarint. This is intentional because:

1. Boundary data dominates storage (written every timestep)
2. Initial fields and mesh are written once
3. The ∼2.7% compression ratio already far exceeds gzip

Auto-detection: The `spaceTimeWindow` BC and `spaceTimeWindowInitCase` automatically detect `.dvz` files and read them correctly. No configuration needed on the reconstruction side.

Encryption (Optional)

Boundary data can be encrypted using X25519 asymmetric encryption (libsodium sealed boxes). This allows extraction with a public key while requiring the private key for reconstruction.

Building with Encryption Support

```
cd src/spaceTimeWindow
export FOAM_USE_SODIUM=1 && wmake

cd applications/utilities/preProcessing/spaceTimeWindowInitCase
export FOAM_USE_SODIUM=1 && wmake

cd ../spaceTimeWindowKeygen
export FOAM_USE_SODIUM=1 && wmake
```

Requires libsodium development package (`libsodium-dev` on Debian/Ubuntu, `libsodium-devel` on RH↩EL/Fedora).

Key Generation

```
spaceTimeWindowKeygen
# Output:
# Public key:  fqzYQ0U8j27tFEr5WzEMylbvXYP+9CAyk0JhwwZ2rwg=
# Private key: QgzxB5b+DGPQH8exbWDe18n4Kv0nu5gqljI2RPBCwl4=
#
# IMPORTANT: Store the private key securely!
```

Encrypted Extraction

Add the public key to the function object configuration:

```
functions
{
    extractSubset
    {
        type            spaceTimeWindowExtract;
        libs            (spaceTimeWindow);

        box             ((0.05 -0.25 0.01) (0.90 0.25 0.38));
        outputDir       "../subset-case";
        fields          (U p nut);

        writeFormat     deltaVarint;
        deltaVarintPrecision  7;

        // Encryption with X25519 public key
        publicKey       "fqzYQ0U8j27tFEr5WzEMylbvXYP+9CAyk0JhwwZ2rwg=";

        writeControl    timeStep;
        writeInterval   1;
    }
}
```

Encrypted files have `.enc` extension (e.g., `U.dvz.enc` for compressed+encrypted).

Decryption During Case Initialization

```
cd subset-case
spaceTimeWindowInitCase -sourceCase ../source-case
# Prompts for private key (no echo):
# Enter private key (base64):
```

The public key is automatically derived from the private key. All encrypted boundary data is decrypted in-place during initialization.

Security Properties

- **Sealed box encryption**: Anonymous sender, only recipient (private key holder) can decrypt
- **X25519 key derivation**: Public key can be computed from private key
- **No unencrypted data on disk**: Encryption happens before writing during extraction
- **Compile-time optional**: Without `FOAM_USE_SODIUM=1`, encryption code is not included

Time Settings Validation

The library enforces identical time settings between extraction and reconstruction to ensure exact timestep matching.

Metadata Storage

During extraction, `spaceTimeWindowExtract` writes an `extractionMetadata` file containing:

```
constant/boundaryData/oldInternalFaces/extractionMetadata
```

Contents:

- `openfoamVersion` - OpenFOAM version string (e.g., "v2512")
- `openfoamApi` - OpenFOAM API number (e.g., 2512)
- `solver` - Application/solver name used during extraction (e.g., "pimpleFoam")
- `deltaT` - Time step used during extraction
- `adjustTimeStep` - Whether adaptive time stepping was enabled
- `timePrecision` - Floating point precision for time values
- `extractionStartTime` - Start time of extraction
- `fixedDeltaT` - Fixed deltaT from controlDict (if specified)

Validation During Reconstruction

The `spaceTimeWindow` BC validates these settings on first use:

1. **OpenFOAM API mismatch** - Warning if reconstruction uses different OpenFOAM version
2. **Solver mismatch** - Warning if reconstruction uses different solver
3. **deltaT mismatch** - Fatal error if reconstruction uses different time step
4. **adjustTimeStep mismatch** - Fatal error if adaptive time stepping setting differs
5. **adjustTimeStep warning** - Warning if both use adaptive time stepping (time steps may not align exactly)

Example error message:

```
FOAM FATAL ERROR:
Time step mismatch between extraction and reconstruction!
    Extraction deltaT: 8.53e-04
    Current deltaT:    1e-03
    Patch: oldInternalFaces

    The spaceTimeWindow BC requires identical time settings
    for exact timestep matching.
    Please set deltaT = 8.53e-04 in system/controlDict
```

Best Practice: Use fixed time stepping (`adjustTimeStep = no`) with identical `deltaT` values for both extraction and reconstruction. All timesteps from the source simulation are required.

Building

```
# Build the library (BC and function object)
cd src/spaceTimeWindow
wmake

# Build the utility
cd applications/utilities/preProcessing/spaceTimeWindowInitCase
wmake
```

Requirements

- **OpenFOAM v2512** (openfoam.com) or compatible ESI-OpenCFD version
- The subset mesh must be created using `subsetMesh` with exact cell extraction (no interpolation)
- For parallel extraction: MPI environment for distributed execution

RANS Compatibility

The library works with transient RANS simulations (e.g., `pimpleFoam` with k-epsilon or k-omega SST):

Supported:

- Transient RANS with time-varying boundary conditions
- All transported turbulence variables (k, epsilon, omega)
- Wall function handling (though subset typically has no walls)

Not supported:

- Steady-state RANS (`simpleFoam`) - the "space-time" concept requires time evolution

Example for k-omega SST:

```
fields    (U p nut k omega);
```

The extraction and reconstruction process is identical to LES. Ensure all transported turbulence quantities are included in the `fields` list.

Example Case

The `examples/ufr2-02` directory contains a complete LES test case:

- **Case**: ERCOFTAC UFR2-02 - Flow around a square cylinder at Re = 22,000
- **Acknowledgment**: Mesh generation script by Niklas Nordin (ERCOFTAC Classic Collection Database Case 043)
- **Reference**: Lyn et al. (1995) J. Fluid Mech. 304, 285-319

```
cd examples/ufr2-02
./Allrun
```

See `examples/ufr2-02/README.md` for detailed instructions.

References

Acknowledging this work

If this software has been useful in your research, please consider:

- Citing: Anton, A.-A. (2011). ∗"Space-Time Window Reconstruction in Parallel High Performance Numeric Simulations. Application for CFD"∗, PhD Thesis, Politehnica University of Timisoara. Available at: https://dspace.upt.ro/jspui/handle/123456789/643

{Anton2011, author = "Alin-Adrian Anton", title = "Space-Time Window Reconstruction in High-Performance Numeric Simulations: Application for {CFD}", school = "Universitatea Politehnica Timișoara", year = "2011", month = "November", type = "PhD thesis", address = "Timișoara, Romania", publisher = "Editura Politehnica", isbn = "978-606-554-390-4", url = "https://dspace.upt.ro/jspui/handle/123456789/643" }

- Or acknowledging: "This work made use of openfoam-spaceTimeWindow https://dev.cs.upt.ro/alin.anton/openfoam-spaceTimeWindow"

Additional references:

- Lyn, D.A., Einav, S., Rodi, W., Park, J.-H. (1995). ∗"A laser-Doppler velocimetry study of ensemble-averaged characteristics of the turbulent near wake of a square cylinder"∗, J. Fluid Mech. 304, 285-319
- ERCOFTAC Classic Collection Database: Case 043

Limitations

- No spatial interpolation - mesh topology must match exactly
- No temporal interpolation - all timesteps from source simulation required
- Boundary data must cover full reconstruction time range
- Steady-state solvers not supported (requires transient simulation)
- For parallel extraction, `reconstructPar` must be run before `spaceTimeWindowInitCase`

2 Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Foam 14

Foam::functionObjects 14

3 Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Foam::deltaVarintCodec 15
fixedValueFvPatchField

Foam::spaceTimeWindowFvPatchField< Type > 32
fvMeshFunctionObject

Foam::functionObjects::spaceTimeWindowExtract 22

Foam::sodiumCrypto 18

4 Class Index

4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Foam::deltaVarintCodec 15

Foam::sodiumCrypto 18

Foam::functionObjects::spaceTimeWindowExtract 22

Foam::spaceTimeWindowFvPatchField< Type > 32

5 File Index

5.1 File List

Here is a list of all files with brief descriptions:

deltaVarintCodec.C 36

deltaVarintCodec.H 36

sodiumCrypto.C 36

sodiumCrypto.H 36

spaceTimeWindowExtract.C 37

spaceTimeWindowExtract.H 39

spaceTimeWindowFvPatchField.C 39

spaceTimeWindowFvPatchField.H 39

spaceTimeWindowFvPatchFields.C 40

spaceTimeWindowFvPatchFields.H 40

spaceTimeWindowInitCase.C 40

spaceTimeWindowKeygen.C 42

6 Namespace Documentation

6.1 Foam Namespace Reference

Namespaces

- functionObjects

Classes

- class deltaVarintCodec
- class sodiumCrypto
- class spaceTimeWindowFvPatchField

Functions

- makePatchTypeFieldTypedefs (spaceTimeWindow)
- makePatchFields (spaceTimeWindow)

6.1.1 Function Documentation

6.1.1.1 makePatchFields()

```
Foam::makePatchFields (
            spaceTimeWindow  )
```

6.1.1.2 makePatchTypeFieldTypedefs()

```
Foam::makePatchTypeFieldTypedefs (
            spaceTimeWindow  )
```

6.2 Foam::functionObjects Namespace Reference

Classes

- class spaceTimeWindowExtract

Functions

- defineTypeNameAndDebug (spaceTimeWindowExtract, 0)
- addToRunTimeSelectionTable (functionObject, spaceTimeWindowExtract, dictionary)

6.2.1 Function Documentation

6.2.1.1 addToRunTimeSelectionTable()

```
Foam::functionObjects::addToRunTimeSelectionTable (
              functionObject ,
              spaceTimeWindowExtract ,
              dictionary  )
```

6.2.1.2 defineTypeNameAndDebug()

```
Foam::functionObjects::defineTypeNameAndDebug (
              spaceTimeWindowExtract ,
              0  )
```

7 Class Documentation

7.1 Foam::deltaVarintCodec Class Reference

```
#include <deltaVarintCodec.H>
```

Static Public Member Functions

- static void write (const fileName &path, const scalarField &field, uint32_t precision=6)
- static void write (const fileName &path, const vectorField &field, uint32_t precision=6)
- static std::vector< uint8_t > encode (const scalarField &field, uint32_t precision=6)
- static std::vector< uint8_t > encode (const vectorField &field, uint32_t precision=6)
- static scalarField readScalar (const fileName &path)
- static vectorField readVector (const fileName &path)
- static scalarField decodeScalar (const std::vector< uint8_t > &buf)
- static vectorField decodeVector (const std::vector< uint8_t > &buf)
- static bool isDeltaVarintFile (const fileName &path)
- static bool isDeltaVarintBuffer (const std::vector< uint8_t > &buf)
- static word fileExtension ()
- static uint32_t magic ()

7.1.1 Detailed Description

Definition at line 72 of file deltaVarintCodec.H.

7.1.2 Member Function Documentation

7.1.2.1 decodeScalar()

```
Foam::scalarField Foam::deltaVarintCodec::decodeScalar (
            const std::vector< uint8_t > & buf )  [static]
```

Definition at line 485 of file deltaVarintCodec.C.

Referenced by decryptBoundaryFile().

7.1.2.2 decodeVector()

```
Foam::vectorField Foam::deltaVarintCodec::decodeVector (
            const std::vector< uint8_t > & buf )  [static]
```

Definition at line 532 of file deltaVarintCodec.C.

Referenced by decryptBoundaryFile().

7.1.2.3 encode() [1/2]

```
std::vector< uint8_t > Foam::deltaVarintCodec::encode (
            const scalarField & field,
            uint32_t precision = 6 )  [static]
```

Definition at line 404 of file deltaVarintCodec.C.

Referenced by Foam::functionObjects::spaceTimeWindowExtract::writeField().

7.1.2.4 encode() [2/2]

```
std::vector< uint8_t > Foam::deltaVarintCodec::encode (
            const vectorField & field,
            uint32_t precision = 6 )  [static]
```

Definition at line 442 of file deltaVarintCodec.C.

7.1.2.5 fileExtension()

```
static word Foam::deltaVarintCodec::fileExtension ( )  [inline], [static]
```

Definition at line 178 of file deltaVarintCodec.H.

Referenced by decryptBoundaryFile(), main(), and Foam::functionObjects::spaceTimeWindowExtract::writeField().

7.1.2.6 isDeltaVarintBuffer()

```
bool Foam::deltaVarintCodec::isDeltaVarintBuffer (
            const std::vector< uint8_t > & buf )  [static]
```

Definition at line 604 of file deltaVarintCodec.C.

Referenced by decryptBoundaryFile().

7.1.2.7 isDeltaVarintFile()

```
bool Foam::deltaVarintCodec::isDeltaVarintFile (
            const fileName & path )  [static]
```

Definition at line 583 of file deltaVarintCodec.C.

7.1.2.8 magic()

```
static uint32_t Foam::deltaVarintCodec::magic ( )  [inline], [static]
```

Definition at line 184 of file deltaVarintCodec.H.

7.1.2.9 readScalar()

```
Foam::scalarField Foam::deltaVarintCodec::readScalar (
            const fileName & path )  [static]
```

Definition at line 273 of file deltaVarintCodec.C.

7.1.2.10 readVector()

```
Foam::vectorField Foam::deltaVarintCodec::readVector (
            const fileName & path )  [static]
```

Definition at line 336 of file deltaVarintCodec.C.

7.1.2.11 write() [1/2]

```
void Foam::deltaVarintCodec::write (
            const fileName & path,
            const scalarField & field,
            uint32_t precision = 6 )  [static]
```

Definition at line 163 of file deltaVarintCodec.C.

Referenced by Foam::functionObjects::spaceTimeWindowExtract::writeField().

7.1.2.12 write() [2/2]

```
void Foam::deltaVarintCodec::write (
            const fileName & path,
            const vectorField & field,
            uint32_t precision = 6 )  [static]
```

Definition at line 216 of file deltaVarintCodec.C.

The documentation for this class was generated from the following files:

- deltaVarintCodec.H
- deltaVarintCodec.C

7.2 Foam::sodiumCrypto Class Reference

```
#include <sodiumCrypto.H>
```

Static Public Member Functions

- static bool available ()
- static bool initialize ()
- static bool generateKeypair (std::vector< uint8_t > &publicKey, std::vector< uint8_t > &privateKey)
- static std::string toBase64 (const std::vector< uint8_t > &data)
- static std::vector< uint8_t > fromBase64 (const std::string &b64)
- static void encryptToFile (const fileName &path, const std::vector< uint8_t > &plaintext, const std::vector< uint8_t > &publicKey)
- static void encryptToFile (const fileName &path, const std::vector< uint8_t > &plaintext, const std::string &publicKeyBase64)
- static std::vector< uint8_t > decryptFromFile (const fileName &path, const std::vector< uint8_t > &public↩ Key, const std::vector< uint8_t > &privateKey)
- static std::vector< uint8_t > decryptFromFile (const fileName &path, const std::string &publicKeyBase64, const std::string &privateKeyBase64)
- static bool isEncryptedFile (const fileName &path)
- static std::string readPrivateKeyFromStdin (const std::string &prompt)
- static std::vector< uint8_t > derivePublicKey (const std::vector< uint8_t > &privateKey)
- static std::string derivePublicKeyBase64 (const std::string &privateKeyBase64)

Static Public Attributes

- static const word fileExtension
- static constexpr uint32_t MAGIC = 0x31434E45
- static constexpr size_t PUBLIC_KEY_SIZE = 32
- static constexpr size_t PRIVATE_KEY_SIZE = 32
- static constexpr size_t HEADER_SIZE = 16

7.2.1 Detailed Description

Definition at line 79 of file sodiumCrypto.H.

7.2.2 Member Function Documentation

7.2.2.1 available()

```
bool Foam::sodiumCrypto::available ( )  [static]
```

Definition at line 33 of file sodiumCrypto.C.

Referenced by main(), and Foam::functionObjects::spaceTimeWindowExtract::read().

7.2.2.2 decryptFromFile() [1/2]

```
std::vector< uint8_t > Foam::sodiumCrypto::decryptFromFile (
            const fileName & path,
            const std::vector< uint8_t > & publicKey,
            const std::vector< uint8_t > & privateKey )  [static]
```

Definition at line 254 of file sodiumCrypto.C.

Referenced by decryptBoundaryFile().

7.2.2.3 decryptFromFile() [2/2]

```
std::vector< uint8_t > Foam::sodiumCrypto::decryptFromFile (
            const fileName & path,
            const std::string & publicKeyBase64,
            const std::string & privateKeyBase64 )  [static]
```

Definition at line 355 of file sodiumCrypto.C.

7.2.2.4 derivePublicKey()

```
std::vector< uint8_t > Foam::sodiumCrypto::derivePublicKey (
            const std::vector< uint8_t > & privateKey )  [static]
```

Definition at line 452 of file sodiumCrypto.C.

Referenced by main().

7.2.2.5 derivePublicKeyBase64()

```
std::string Foam::sodiumCrypto::derivePublicKeyBase64 (
            const std::string & privateKeyBase64 )  [static]
```

Definition at line 489 of file sodiumCrypto.C.

7.2.2.6 encryptToFile() [1/2]

```
void Foam::sodiumCrypto::encryptToFile (
            const fileName & path,
            const std::vector< uint8_t > & plaintext,
            const std::vector< uint8_t > & publicKey )  [static]
```

Definition at line 163 of file sodiumCrypto.C.

Referenced by Foam::functionObjects::spaceTimeWindowExtract::writeField().

7.2.2.7 encryptToFile() [2/2]

```
void Foam::sodiumCrypto::encryptToFile (
            const fileName & path,
            const std::vector< uint8_t > & plaintext,
            const std::string & publicKeyBase64 )  [static]
```

Definition at line 234 of file sodiumCrypto.C.

7.2.2.8 fromBase64()

```
std::vector< uint8_t > Foam::sodiumCrypto::fromBase64 (
            const std::string & b64 )  [static]
```

Definition at line 123 of file sodiumCrypto.C.

Referenced by main(), Foam::functionObjects::spaceTimeWindowExtract::read(), and Foam::functionObjects↩
::spaceTimeWindowExtract::writeField().

7.2.2.9 generateKeypair()

```
bool Foam::sodiumCrypto::generateKeypair (
            std::vector< uint8_t > & publicKey,
            std::vector< uint8_t > & privateKey )  [static]
```

Definition at line 63 of file sodiumCrypto.C.

Referenced by main().

7.2.2.10 initialize()

```
bool Foam::sodiumCrypto::initialize ( )  [static]
```

Definition at line 43 of file sodiumCrypto.C.

7.2.2.11 isEncryptedFile()

```
bool Foam::sodiumCrypto::isEncryptedFile (
            const fileName & path )  [static]
```

Definition at line 382 of file sodiumCrypto.C.

7.2.2.12 readPrivateKeyFromStdin()

```
std::string Foam::sodiumCrypto::readPrivateKeyFromStdin (
            const std::string & prompt )  [static]
```

Definition at line 409 of file sodiumCrypto.C.

Referenced by main().

7.2.2.13 toBase64()

```
std::string Foam::sodiumCrypto::toBase64 (
            const std::vector< uint8_t > & data )  [static]
```

Definition at line 89 of file sodiumCrypto.C.

Referenced by main().

7.2.3 Member Data Documentation

7.2.3.1 fileExtension

```
const Foam::word Foam::sodiumCrypto::fileExtension  [static]
```

Definition at line 86 of file sodiumCrypto.H.

Referenced by decryptBoundaryFile(), main(), and Foam::functionObjects::spaceTimeWindowExtract::writeField().

7.2.3.2 HEADER_SIZE

```
constexpr size_t Foam::sodiumCrypto::HEADER_SIZE = 16  [static]
```

Definition at line 98 of file sodiumCrypto.H.

7.2.3.3 MAGIC

```
constexpr uint32_t Foam::sodiumCrypto::MAGIC = 0x31434E45  [static]
```

Definition at line 89 of file sodiumCrypto.H.

7.2.3.4 PRIVATE_KEY_SIZE

```
constexpr size_t Foam::sodiumCrypto::PRIVATE_KEY_SIZE = 32  [static]
```

Definition at line 95 of file sodiumCrypto.H.

Referenced by main().

7.2.3.5 PUBLIC_KEY_SIZE

```
constexpr size_t Foam::sodiumCrypto::PUBLIC_KEY_SIZE = 32  [static]
```

Definition at line 92 of file sodiumCrypto.H.

Referenced by Foam::functionObjects::spaceTimeWindowExtract::read().

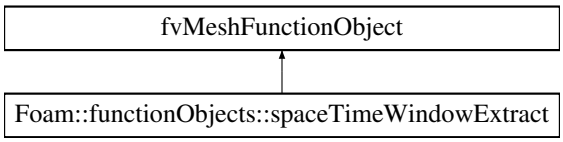
The documentation for this class was generated from the following files:

- sodiumCrypto.H
- sodiumCrypto.C

7.3 Foam::functionObjects::spaceTimeWindowExtract Class Reference

```
#include <spaceTimeWindowExtract.H>
```

Inheritance diagram for Foam::functionObjects::spaceTimeWindowExtract:



Public Member Functions

- TypeName ("spaceTimeWindowExtract")
- spaceTimeWindowExtract (const word &name, const Time &runTime, const dictionary &dict)
- virtual ~spaceTimeWindowExtract ()
- virtual bool read (const dictionary &dict)
- virtual bool execute ()
- virtual bool write ()
- template<class Type >
  Foam::tmp< Foam::Field< Type > > interpolateToFaces (const GeometricField< Type, fvPatchField, vol↩
  Mesh > &vf) const
- template<class Type >
  Foam::tmp< Foam::Field< Type > > gatherFieldToMaster (const Field< Type > &localField) const

Protected Member Functions

- void writeFoamFileHeader (Ostream &os, const word &className, const word &objectName) const
- void initializeSubset ()
- void initializeParallelComm ()
- void identifyProcessorBoundaryFaces (const boundBox &bb, const labelList &cellsInBox)
- void gatherFaceDataToMaster (vectorField &allFaceCentres, scalarField &allFaceWeights, labelList &all↩ FaceCellsInside, labelList &allFaceCellsOutside) const
- void gatherAndWriteSubsetMesh ()
- void gatherAndWriteInitialFields ()
- void writeSubsetMesh ()
- void writeInitialFields ()
- void writeBoundaryData ()
- void updateMetadataTimesteps ()
- template<class Type >
  tmp< Field< Type > > interpolateToFaces (const GeometricField< Type, fvPatchField, volMesh > &vf) const
- template<class Type >
  tmp< Field< Type > > gatherFieldToMaster (const Field< Type > &localField) const
- template<class Type >
  void writeField (const fileName &dir, const word &fieldName, const Field< Type > &field) const

Protected Attributes

- point boxMin_
- point boxMax_
- fileName outputDir_
- wordList fieldNames_
- IOstreamOption::streamFormat writeFormat_
- IOstreamOption::compressionType writeCompression_
- bool useDeltaVarint_
- label deltaVarintPrecision_
- bool useEncryption_
- string publicKeyBase64_
- autoPtr< fvMeshSubset > meshSubsetPtr_
- bool subsetInitialized_
- bool meshWritten_
- labelList oldInternalFaceIndices_
- labelList faceCellsInside_
- labelList faceCellsOutside_
- scalarList faceWeights_
- vectorField faceCentres_
- wordList extractedTimesteps_
- autoPtr< globalIndex > globalFaceIndexPtr_
- label nGlobalFaces_
- autoPtr< mapDistribute > outsideCellMapPtr_
- labelList remoteFaceIndices_
- labelList remoteCellIndices_
- labelList remoteCellProcs_
- bool hasProcessorBoundaryFaces_

7.3.1 Detailed Description

Definition at line 106 of file spaceTimeWindowExtract.H.

7.3.2 Constructor & Destructor Documentation

7.3.2.1 spaceTimeWindowExtract()

```
Foam::functionObjects::spaceTimeWindowExtract::spaceTimeWindowExtract (
            const word & name,
            const Time & runTime,
            const dictionary & dict )
```

Definition at line 1570 of file spaceTimeWindowExtract.C.

7.3.2.2 ~spaceTimeWindowExtract()

```
Foam::functionObjects::spaceTimeWindowExtract::~spaceTimeWindowExtract ( )  [virtual]
```

Definition at line 1557 of file spaceTimeWindowExtract.C.

7.3.3 Member Function Documentation

7.3.3.1 execute()

```
bool Foam::functionObjects::spaceTimeWindowExtract::execute ( )  [virtual]
```

Definition at line 1744 of file spaceTimeWindowExtract.C.

7.3.3.2 gatherAndWriteInitialFields()

```
void Foam::functionObjects::spaceTimeWindowExtract::gatherAndWriteInitialFields ( )  [protected]
```

Definition at line 630 of file spaceTimeWindowExtract.C.

7.3.3.3 gatherAndWriteSubsetMesh()

```
void Foam::functionObjects::spaceTimeWindowExtract::gatherAndWriteSubsetMesh ( )  [protected]
```

Definition at line 588 of file spaceTimeWindowExtract.C.

7.3.3.4 gatherFaceDataToMaster()

```
void Foam::functionObjects::spaceTimeWindowExtract::gatherFaceDataToMaster (
            vectorField & allFaceCentres,
            scalarField & allFaceWeights,
            labelList & allFaceCellsInside,
            labelList & allFaceCellsOutside ) const  [protected]
```

Definition at line 524 of file spaceTimeWindowExtract.C.

7.3.3.5 gatherFieldToMaster() [1/2]

```
template<class Type >
tmp<Field<Type> > Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToMaster (
            const Field< Type > & localField ) const  [protected]
```

7.3.3.6 gatherFieldToMaster() [2/2]

```
template<class Type >
Foam::tmp<Foam::Field<Type> > Foam::functionObjects::spaceTimeWindowExtract::gatherFieldTo↩
Master (
            const Field< Type > & localField ) const
```

Definition at line 1410 of file spaceTimeWindowExtract.C.

7.3.3.7 identifyProcessorBoundaryFaces()

```
void Foam::functionObjects::spaceTimeWindowExtract::identifyProcessorBoundaryFaces (
            const boundBox & bb,
            const labelList & cellsInBox )  [protected]
```

Definition at line 513 of file spaceTimeWindowExtract.C.

Referenced by initializeSubset().

7.3.3.8 initializeParallelComm()

```
void Foam::functionObjects::spaceTimeWindowExtract::initializeParallelComm ( )  [protected]
```

Definition at line 505 of file spaceTimeWindowExtract.C.

Referenced by initializeSubset().

7.3.3.9 initializeSubset()

```
void Foam::functionObjects::spaceTimeWindowExtract::initializeSubset ( )  [protected]
```

Definition at line 74 of file spaceTimeWindowExtract.C.

References boxMax_, boxMin_, faceCellsInside_, faceCellsOutside_, faceCentres_, faceWeights_, globalFaceIndexPtr_, hasProcessorBoundaryFaces_, identifyProcessorBoundaryFaces(), initializeParallelComm(), meshSubsetPtr_, nGlobalFaces_, oldInternalFaceIndices_, remoteCellIndices_, remoteCellProcs_, remoteFaceIndices_, and subsetInitialized_.

7.3.3.10 interpolateToFaces() [1/2]

```
template<class Type >
tmp<Field<Type> > Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces (
            const GeometricField< Type, fvPatchField, volMesh > & vf ) const  [protected]
```

7.3.3.11 interpolateToFaces() [2/2]

```
template<class Type >
Foam::tmp<Foam::Field<Type> > Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces (
            const GeometricField< Type, fvPatchField, volMesh > & vf ) const
```

Definition at line 1298 of file spaceTimeWindowExtract.C.

7.3.3.12 read()

```
bool Foam::functionObjects::spaceTimeWindowExtract::read (
            const dictionary & dict )  [virtual]
```

Definition at line 1610 of file spaceTimeWindowExtract.C.

References Foam::sodiumCrypto::available(), Foam::sodiumCrypto::fromBase64(), and Foam::sodiumCrypto::PUBLIC_KEY_SIZE.

7.3.3.13 TypeName()

```
Foam::functionObjects::spaceTimeWindowExtract::TypeName (
            "spaceTimeWindowExtract"  )
```

7.3.3.14 updateMetadataTimesteps()

```
void Foam::functionObjects::spaceTimeWindowExtract::updateMetadataTimesteps ( )  [protected]
```

Definition at line 1273 of file spaceTimeWindowExtract.C.

7.3.3.15 write()

```
bool Foam::functionObjects::spaceTimeWindowExtract::write ( )  [virtual]
```

Definition at line 1784 of file spaceTimeWindowExtract.C.

7.3.3.16 writeBoundaryData()

```
void Foam::functionObjects::spaceTimeWindowExtract::writeBoundaryData ( )  [protected]
```

Definition at line 1196 of file spaceTimeWindowExtract.C.

7.3.3.17 writeField()

```
template<class Type >
void Foam::functionObjects::spaceTimeWindowExtract::writeField (
            const fileName & dir,
            const word & fieldName,
            const Field< Type > & field ) const  [protected]
```

Definition at line 1456 of file spaceTimeWindowExtract.C.

References Foam::deltaVarintCodec::encode(), Foam::sodiumCrypto::encryptToFile(), Foam::sodiumCrypto::file↩Extension, Foam::deltaVarintCodec::fileExtension(), Foam::sodiumCrypto::fromBase64(), and Foam::deltaVarint↩Codec::write().

7.3.3.18 writeFoamFileHeader()

```
void Foam::functionObjects::spaceTimeWindowExtract::writeFoamFileHeader (
            Ostream & os,
            const word & className,
            const word & objectName ) const  [protected]
```

Definition at line 56 of file spaceTimeWindowExtract.C.

7.3.3.19 writeInitialFields()

```
void Foam::functionObjects::spaceTimeWindowExtract::writeInitialFields ( )  [protected]
```

Definition at line 1066 of file spaceTimeWindowExtract.C.

7.3.3.20 writeSubsetMesh()

```
void Foam::functionObjects::spaceTimeWindowExtract::writeSubsetMesh ( )  [protected]
```

Definition at line 870 of file spaceTimeWindowExtract.C.

7.3.4 Member Data Documentation

7.3.4.1 boxMax_

```
point Foam::functionObjects::spaceTimeWindowExtract::boxMax_  [protected]
```

Definition at line 118 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.2 boxMin_

```
point Foam::functionObjects::spaceTimeWindowExtract::boxMin_  [protected]
```

Definition at line 115 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.3 deltaVarintPrecision_

```
label Foam::functionObjects::spaceTimeWindowExtract::deltaVarintPrecision_  [protected]
```

Definition at line 136 of file spaceTimeWindowExtract.H.

7.3.4.4 extractedTimesteps_

```
wordList Foam::functionObjects::spaceTimeWindowExtract::extractedTimesteps_  [protected]
```

Definition at line 169 of file spaceTimeWindowExtract.H.

7.3.4.5 faceCellsInside_

```
labelList Foam::functionObjects::spaceTimeWindowExtract::faceCellsInside_ [protected]
```

Definition at line 157 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.6 faceCellsOutside_

```
labelList Foam::functionObjects::spaceTimeWindowExtract::faceCellsOutside_ [protected]
```

Definition at line 160 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.7 faceCentres_

```
vectorField Foam::functionObjects::spaceTimeWindowExtract::faceCentres_ [protected]
```

Definition at line 166 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.8 faceWeights_

```
scalarList Foam::functionObjects::spaceTimeWindowExtract::faceWeights_ [protected]
```

Definition at line 163 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.9 fieldNames_

```
wordList Foam::functionObjects::spaceTimeWindowExtract::fieldNames_ [protected]
```

Definition at line 124 of file spaceTimeWindowExtract.H.

7.3.4.10 globalFaceIndexPtr_

```
autoPtr<globalIndex> Foam::functionObjects::spaceTimeWindowExtract::globalFaceIndexPtr_↩
[protected]
```

Definition at line 174 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.11 hasProcessorBoundaryFaces_

```
bool Foam::functionObjects::spaceTimeWindowExtract::hasProcessorBoundaryFaces_  [protected]
```

Definition at line 193 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.12 meshSubsetPtr_

```
autoPtr<fvMeshSubset> Foam::functionObjects::spaceTimeWindowExtract::meshSubsetPtr_  [protected]
```

Definition at line 145 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.13 meshWritten_

```
bool Foam::functionObjects::spaceTimeWindowExtract::meshWritten_  [protected]
```

Definition at line 151 of file spaceTimeWindowExtract.H.

7.3.4.14 nGlobalFaces_

```
label Foam::functionObjects::spaceTimeWindowExtract::nGlobalFaces_  [protected]
```

Definition at line 177 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.15 oldInternalFaceIndices_

```
labelList Foam::functionObjects::spaceTimeWindowExtract::oldInternalFaceIndices_  [protected]
```

Definition at line 154 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.16 outputDir_

```
fileName Foam::functionObjects::spaceTimeWindowExtract::outputDir_  [protected]
```

Definition at line 121 of file spaceTimeWindowExtract.H.

7.3.4.17 outsideCellMapPtr_

```
autoPtr<mapDistribute> Foam::functionObjects::spaceTimeWindowExtract::outsideCellMapPtr_↩
[protected]
```

Definition at line 181 of file spaceTimeWindowExtract.H.

7.3.4.18 publicKeyBase64_

```
string Foam::functionObjects::spaceTimeWindowExtract::publicKeyBase64_  [protected]
```

Definition at line 142 of file spaceTimeWindowExtract.H.

7.3.4.19 remoteCellIndices_

```
labelList Foam::functionObjects::spaceTimeWindowExtract::remoteCellIndices_  [protected]
```

Definition at line 187 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.20 remoteCellProcs_

```
labelList Foam::functionObjects::spaceTimeWindowExtract::remoteCellProcs_  [protected]
```

Definition at line 190 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.21 remoteFaceIndices_

```
labelList Foam::functionObjects::spaceTimeWindowExtract::remoteFaceIndices_  [protected]
```

Definition at line 184 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.22 subsetInitialized_

```
bool Foam::functionObjects::spaceTimeWindowExtract::subsetInitialized_  [protected]
```

Definition at line 148 of file spaceTimeWindowExtract.H.

Referenced by initializeSubset().

7.3.4.23 useDeltaVarint_

```
bool Foam::functionObjects::spaceTimeWindowExtract::useDeltaVarint_  [protected]
```

Definition at line 133 of file spaceTimeWindowExtract.H.

7.3.4.24 useEncryption_

```
bool Foam::functionObjects::spaceTimeWindowExtract::useEncryption_  [protected]
```

Definition at line 139 of file spaceTimeWindowExtract.H.

7.3.4.25 writeCompression_

```
IOstreamOption::compressionType Foam::functionObjects::spaceTimeWindowExtract::writeCompression↩
_  [protected]
```

Definition at line 130 of file spaceTimeWindowExtract.H.

7.3.4.26 writeFormat_

```
IOstreamOption::streamFormat Foam::functionObjects::spaceTimeWindowExtract::writeFormat_↩
[protected]
```

Definition at line 127 of file spaceTimeWindowExtract.H.

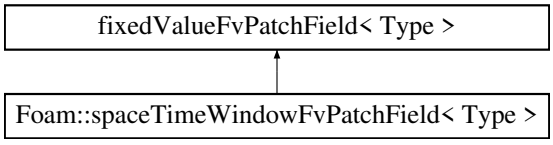
The documentation for this class was generated from the following files:

- spaceTimeWindowExtract.H
- spaceTimeWindowExtract.C

7.4 Foam::spaceTimeWindowFvPatchField< Type > Class Template Reference

```
#include <spaceTimeWindowFvPatchField.H>
```

Inheritance diagram for Foam::spaceTimeWindowFvPatchField< Type >:



Public Member Functions

- TypeName ("spaceTimeWindow")
- spaceTimeWindowFvPatchField (const fvPatch &, const DimensionedField< Type, volMesh > &)
- spaceTimeWindowFvPatchField (const fvPatch &, const DimensionedField< Type, volMesh > &, const dictionary &)
- spaceTimeWindowFvPatchField (const spaceTimeWindowFvPatchField< Type > &, const fvPatch &, const DimensionedField< Type, volMesh > &, const fvPatchFieldMapper &)
- spaceTimeWindowFvPatchField (const spaceTimeWindowFvPatchField< Type > &)
- virtual tmp< fvPatchField< Type > > clone () const
- spaceTimeWindowFvPatchField (const spaceTimeWindowFvPatchField< Type > &, const DimensionedField< Type, volMesh > &)
- virtual tmp< fvPatchField< Type > > clone (const DimensionedField< Type, volMesh > &iF) const
- virtual void autoMap (const fvPatchFieldMapper &)
- virtual void rmap (const fvPatchField< Type > &, const labelList &)
- virtual void updateCoeffs ()
- virtual bool fixesValue () const
- virtual void write (Ostream &) const

7.4.1 Detailed Description

template<class Type>
class Foam::spaceTimeWindowFvPatchField< Type >

Definition at line 95 of file spaceTimeWindowFvPatchField.H.

7.4.2 Constructor & Destructor Documentation

7.4.2.1 spaceTimeWindowFvPatchField() [1/5]

```
template<class Type >
Foam::spaceTimeWindowFvPatchField< Type >::spaceTimeWindowFvPatchField (
            const fvPatch & p,
            const DimensionedField< Type, volMesh > & iF )
```

Definition at line 420 of file spaceTimeWindowFvPatchField.C.

7.4.2.2 spaceTimeWindowFvPatchField() [2/5]

```
template<class Type >
Foam::spaceTimeWindowFvPatchField< Type >::spaceTimeWindowFvPatchField (
            const fvPatch & p,
            const DimensionedField< Type, volMesh > & iF,
            const dictionary & dict )
```

Definition at line 442 of file spaceTimeWindowFvPatchField.C.

7.4.2.3 spaceTimeWindowFvPatchField() [3/5]

```
template<class Type >
Foam::spaceTimeWindowFvPatchField< Type >::spaceTimeWindowFvPatchField (
            const spaceTimeWindowFvPatchField< Type > & ptf,
            const fvPatch & p,
            const DimensionedField< Type, volMesh > & iF,
            const fvPatchFieldMapper & mapper )
```

Definition at line 473 of file spaceTimeWindowFvPatchField.C.

7.4.2.4 spaceTimeWindowFvPatchField() [4/5]

```
template<class Type >
Foam::spaceTimeWindowFvPatchField< Type >::spaceTimeWindowFvPatchField (
            const spaceTimeWindowFvPatchField< Type > & ptf )
```

Definition at line 497 of file spaceTimeWindowFvPatchField.C.

7.4.2.5 spaceTimeWindowFvPatchField() [5/5]

```
template<class Type >
Foam::spaceTimeWindowFvPatchField< Type >::spaceTimeWindowFvPatchField (
            const spaceTimeWindowFvPatchField< Type > & ptf,
            const DimensionedField< Type, volMesh > & iF )
```

Definition at line 518 of file spaceTimeWindowFvPatchField.C.

7.4.3 Member Function Documentation

7.4.3.1 autoMap()

```
template<class Type >
void Foam::spaceTimeWindowFvPatchField< Type >::autoMap (
            const fvPatchFieldMapper & m )  [virtual]
```

Definition at line 542 of file spaceTimeWindowFvPatchField.C.

7.4.3.2 clone() [1/2]

```
template<class Type>
virtual tmp<fvPatchField<Type> > Foam::spaceTimeWindowFvPatchField< Type >::clone ( ) const
[inline], [virtual]
```

Definition at line 190 of file spaceTimeWindowFvPatchField.H.

7.4.3.3 clone() [2/2]

```
template<class Type>
virtual tmp<fvPatchField<Type> > Foam::spaceTimeWindowFvPatchField< Type >::clone (
            const DimensionedField< Type, volMesh > & iF ) const  [inline], [virtual]
```

Definition at line 207 of file spaceTimeWindowFvPatchField.H.

7.4.3.4 fixesValue()

```
template<class Type>
virtual bool Foam::spaceTimeWindowFvPatchField< Type >::fixesValue ( ) const  [inline], [virtual]
```

Definition at line 246 of file spaceTimeWindowFvPatchField.H.

7.4.3.5 rmap()

```
template<class Type >
void Foam::spaceTimeWindowFvPatchField< Type >::rmap (
            const fvPatchField< Type > & ptf,
            const labelList & addr )  [virtual]
```

Definition at line 558 of file spaceTimeWindowFvPatchField.C.

7.4.3.6 TypeName()

```
template<class Type>
Foam::spaceTimeWindowFvPatchField< Type >::TypeName (
            "spaceTimeWindow"  )
```

7.4.3.7 updateCoeffs()

```
template<class Type >
void Foam::spaceTimeWindowFvPatchField< Type >::updateCoeffs ( )  [virtual]
```

Definition at line 574 of file spaceTimeWindowFvPatchField.C.

7.4.3.8 write()

```
template<class Type >
void Foam::spaceTimeWindowFvPatchField< Type >::write (
            Ostream & os ) const  [virtual]
```

Definition at line 620 of file spaceTimeWindowFvPatchField.C.

The documentation for this class was generated from the following files:

- spaceTimeWindowFvPatchField.H
- spaceTimeWindowFvPatchField.C

8 File Documentation

8.1 deltaVarintCodec.C File Reference

```
#include "deltaVarintCodec.H"
#include "error.H"
#include <cmath>
#include <cstring>
```

8.2 deltaVarintCodec.H File Reference

```
#include "scalarField.H"
#include "vectorField.H"
#include "fileName.H"
#include <fstream>
#include <vector>
#include <cstdint>
```

Classes

- class Foam::deltaVarintCodec

Namespaces

- Foam

8.3 README.md File Reference

8.4 sodiumCrypto.C File Reference

```
#include "sodiumCrypto.H"
#include "error.H"
#include <fstream>
#include <cstring>
#include <sodium.h>
#include <termios.h>
#include <unistd.h>
```

8.5 sodiumCrypto.H File Reference

```
#include "fileName.H"
#include "word.H"
#include <vector>
#include <string>
#include <cstdint>
```

Classes

- class Foam::sodiumCrypto

Namespaces

- Foam

8.6 spaceTimeWindowExtract.C File Reference

```
#include "spaceTimeWindowExtract.H"
#include "addToRunTimeSelectionTable.H"
#include "fvMeshSubset.H"
#include "OFstream.H"
#include "Time.H"
#include "surfaceFields.H"
#include "Pstream.H"
#include "PstreamBuffers.H"
#include "foamVersion.H"
#include "argList.H"
#include "boundBox.H"
#include "polyPatch.H"
#include "processorPolyPatch.H"
#include "Tuple2.H"
#include "globalIndex.H"
#include "ListOps.H"
#include "deltaVarintCodec.H"
#include "sodiumCrypto.H"
#include <fstream>
#include <type_traits>
#include <cstring>
```

Namespaces

- Foam
- Foam::functionObjects

Functions

- Foam::functionObjects::defineTypeNameAndDebug (spaceTimeWindowExtract, 0)
- Foam::functionObjects::addToRunTimeSelectionTable (functionObject, spaceTimeWindowExtract, dictionary)
- template Foam::tmp< Foam::Field< Foam::scalar > > Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces< (const GeometricField< scalar, fvPatchField, volMesh > &) const
- template Foam::tmp< Foam::Field< Foam::vector > > Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces< (const GeometricField< vector, fvPatchField, volMesh > &) const
- template Foam::tmp< Foam::Field< Foam::scalar > > Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToMaste (const Field< scalar > &) const
- template Foam::tmp< Foam::Field< Foam::vector > > Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToMaste (const Field< vector > &) const
- template void Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::scalar > (const fileName &, const word &, const Field< scalar > &) const
- template void Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::vector > (const fileName &, const word &, const Field< vector > &) const

8.6.1 Function Documentation

8.6.1.1 Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToMaster< Foam::scalar >()

```
template Foam::tmp<Foam::Field<Foam::scalar> > Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToM
Foam::scalar > (
            const Field< scalar > &  ) const
```

8.6.1.2 Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToMaster< Foam::vector >()

```
template Foam::tmp<Foam::Field<Foam::vector> > Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToM
Foam::vector > (
            const Field< vector > &  ) const
```

8.6.1.3 Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces< Foam::scalar >()

```
template Foam::tmp<Foam::Field<Foam::scalar> > Foam::functionObjects::spaceTimeWindowExtract::interpolateToF
Foam::scalar > (
            const GeometricField< scalar, fvPatchField, volMesh > &  ) const
```

8.6.1.4 Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces< Foam::vector >()

```
template Foam::tmp<Foam::Field<Foam::vector> > Foam::functionObjects::spaceTimeWindowExtract::interpolateToF
Foam::vector > (
            const GeometricField< vector, fvPatchField, volMesh > &  ) const
```

8.6.1.5 Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::scalar >()

```
template void Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::scalar > (
            const fileName & ,
            const word & ,
            const Field< scalar > &  ) const
```

8.6.1.6 Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::vector >()

```
template void Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::vector > (
            const fileName & ,
            const word & ,
            const Field< vector > &  ) const
```

8.7 spaceTimeWindowExtract.H File Reference

```
#include "fvMeshFunctionObject.H"
#include "fvMeshSubset.H"
#include "volFields.H"
#include "globalIndex.H"
#include "mapDistribute.H"
#include "IOstreamOption.H"
```

Classes

- class Foam::functionObjects::spaceTimeWindowExtract

Namespaces

- Foam
- Foam::functionObjects

8.8 spaceTimeWindowFvPatchField.C File Reference

```
#include "spaceTimeWindowFvPatchField.H"
#include "Time.H"
#include "IFstream.H"
#include "dictionary.H"
#include "foamVersion.H"
#include "token.H"
#include "IOstreamOption.H"
#include "deltaVarintCodec.H"
#include "pTraits.H"
#include <type_traits>
```

8.9 spaceTimeWindowFvPatchField.H File Reference

```
#include "fixedValueFvPatchFields.H"
#include "instantList.H"
```

Classes

- class Foam::spaceTimeWindowFvPatchField< Type >

Namespaces

- Foam

8.10 spaceTimeWindowFvPatchFields.C File Reference

```
#include "spaceTimeWindowFvPatchField.H"
#include "addToRunTimeSelectionTable.H"
#include "volFields.H"
```

Namespaces

- Foam

Functions

- Foam::makePatchTypeFieldTypedefs (spaceTimeWindow)
- Foam::makePatchFields (spaceTimeWindow)

8.11 spaceTimeWindowFvPatchFields.H File Reference

```
#include "spaceTimeWindowFvPatchField.H"
#include "fieldTypes.H"
```

Namespaces

- Foam

Functions

- Foam::makePatchTypeFieldTypedefs (spaceTimeWindow)

8.12 spaceTimeWindowInitCase.C File Reference

```
#include "argList.H"
#include "Time.H"
#include "fvMesh.H"
#include "fvMeshSubset.H"
#include "IOdictionary.H"
#include "IFstream.H"
#include "OFstream.H"
#include "volFields.H"
#include "surfaceFields.H"
#include "OSspecific.H"
#include "foamVersion.H"
#include "Tuple2.H"
#include "DynamicList.H"
#include "SortableList.H"
#include "boundBox.H"
#include "deltaVarintCodec.H"
#include "sodiumCrypto.H"
#include <fstream>
#include <sstream>
#include "setRootCase.H"
```

Functions

- void copyFile (const fileName &src, const fileName &dst)
- void copyDirectory (const fileName &src, const fileName &dst)
- word decryptBoundaryFile (const fileName &encryptedPath, const fileName &outputDir, const std::vector< uint8_t > &publicKey, const std::vector< uint8_t > &privateKey, bool verbose=true)
- void writeControlDict (const fileName &targetDir, const dictionary &metadata, const dictionary &sourceControlDict, const word &startTimeName, const word &endTimeName)
- void writeInitialField (const fileName &targetDir, const word &fieldName, const word &fieldType, const dictionary &fieldDict, const word &timeName, const polyBoundaryMesh &bMesh)
- int main (int argc, char *argv[])

8.12.1 Function Documentation

8.12.1.1 copyDirectory()

```
void copyDirectory (
            const fileName & src,
            const fileName & dst )
```

Definition at line 77 of file spaceTimeWindowInitCase.C.

8.12.1.2 copyFile()

```
void copyFile (
            const fileName & src,
            const fileName & dst )
```

Definition at line 63 of file spaceTimeWindowInitCase.C.

Referenced by main().

8.12.1.3 decryptBoundaryFile()

```
word decryptBoundaryFile (
            const fileName & encryptedPath,
            const fileName & outputDir,
            const std::vector< uint8_t > & publicKey,
            const std::vector< uint8_t > & privateKey,
            bool verbose = true )
```

Definition at line 90 of file spaceTimeWindowInitCase.C.

References Foam::deltaVarintCodec::decodeScalar(), Foam::deltaVarintCodec::decodeVector(), Foam::sodiumCrypto::decryptFromFile(), Foam::sodiumCrypto::fileExtension, Foam::deltaVarintCodec::fileExtension(), and Foam::deltaVarintCodec::isDeltaVarintBuffer().

Referenced by main().

8.12.1.4 main()

```
int main (
            int argc,
            char * argv[] )
```

Definition at line 579 of file spaceTimeWindowInitCase.C.

References Foam::sodiumCrypto::available(), copyFile(), decryptBoundaryFile(), Foam::sodiumCrypto::derive↩PublicKey(), Foam::sodiumCrypto::fileExtension, Foam::deltaVarintCodec::fileExtension(), Foam::sodiumCrypto↩::fromBase64(), Foam::sodiumCrypto::PRIVATE_KEY_SIZE, Foam::sodiumCrypto::readPrivateKeyFromStdin(), and writeControlDict().

8.12.1.5 writeControlDict()

```
void writeControlDict (
            const fileName & targetDir,
            const dictionary & metadata,
            const dictionary & sourceControlDict,
            const word & startTimeName,
            const word & endTimeName )
```

Definition at line 291 of file spaceTimeWindowInitCase.C.

Referenced by main().

8.12.1.6 writeInitialField()

```
void writeInitialField (
            const fileName & targetDir,
            const word & fieldName,
            const word & fieldType,
            const dictionary & fieldDict,
            const word & timeName,
            const polyBoundaryMesh & bMesh )
```

Definition at line 391 of file spaceTimeWindowInitCase.C.

8.13 spaceTimeWindowKeygen.C File Reference

```
#include "argList.H"
#include "sodiumCrypto.H"
#include <iostream>
#include "setRootCase.H"
```

Functions

- int main (int argc, char ∗argv[ ])

8.13.1 Function Documentation

8.13.1.1 main()

```
int main (
            int argc,
            char * argv[] )
```

Definition at line 43 of file spaceTimeWindowKeygen.C.

References Foam::sodiumCrypto::available(), Foam::sodiumCrypto::generateKeypair(), and Foam::sodium↩ Crypto::toBase64().

Index

~spaceTimeWindowExtract
Foam::functionObjects::spaceTimeWindowExtract, 24

addToRunTimeSelectionTable
Foam::functionObjects, 15
autoMap
Foam::spaceTimeWindowFvPatchField, 34
available
Foam::sodiumCrypto, 19

boxMax_
Foam::functionObjects::spaceTimeWindowExtract, 28
boxMin_
Foam::functionObjects::spaceTimeWindowExtract, 28

clone
Foam::spaceTimeWindowFvPatchField, 34
copyDirectory
spaceTimeWindowInitCase.C, 41
copyFile
spaceTimeWindowInitCase.C, 41

decodeScalar
Foam::deltaVarintCodec, 16
decodeVector
Foam::deltaVarintCodec, 16
decryptBoundaryFile
spaceTimeWindowInitCase.C, 41
decryptFromFile
Foam::sodiumCrypto, 19
defineTypeNameAndDebug
Foam::functionObjects, 15
deltaVarintCodec.C, 36
deltaVarintCodec.H, 36
deltaVarintPrecision_
Foam::functionObjects::spaceTimeWindowExtract, 28
derivePublicKey
Foam::sodiumCrypto, 19
derivePublicKeyBase64
Foam::sodiumCrypto, 19

encode
Foam::deltaVarintCodec, 16
encryptToFile
Foam::sodiumCrypto, 19, 20
execute
Foam::functionObjects::spaceTimeWindowExtract, 24
extractedTimesteps_
Foam::functionObjects::spaceTimeWindowExtract, 28

faceCellsInside_
Foam::functionObjects::spaceTimeWindowExtract, 28
faceCellsOutside_
Foam::functionObjects::spaceTimeWindowExtract, 29
faceCentres_
Foam::functionObjects::spaceTimeWindowExtract, 29
faceWeights_
Foam::functionObjects::spaceTimeWindowExtract, 29
fieldNames_
Foam::functionObjects::spaceTimeWindowExtract, 29
fileExtension
Foam::deltaVarintCodec, 16
Foam::sodiumCrypto, 21
fixesValue
Foam::spaceTimeWindowFvPatchField, 35
Foam, 14
makePatchFields, 14
makePatchTypeFieldTypedefs, 14
Foam::deltaVarintCodec, 15
decodeScalar, 16
decodeVector, 16
encode, 16
fileExtension, 16
isDeltaVarintBuffer, 16
isDeltaVarintFile, 17
magic, 17
readScalar, 17
readVector, 17
write, 17
Foam::functionObjects, 14
addToRunTimeSelectionTable, 15
defineTypeNameAndDebug, 15
Foam::functionObjects::spaceTimeWindowExtract, 22
~spaceTimeWindowExtract, 24
boxMax_, 28
boxMin_, 28
deltaVarintPrecision_, 28
execute, 24
extractedTimesteps_, 28
faceCellsInside_, 28
faceCellsOutside_, 29
faceCentres_, 29
faceWeights_, 29
fieldNames_, 29
gatherAndWriteInitialFields, 24
gatherAndWriteSubsetMesh, 24
gatherFaceDataToMaster, 24
gatherFieldToMaster, 25
globalFaceIndexPtr_, 29
hasProcessorBoundaryFaces_, 29
identifyProcessorBoundaryFaces, 25

initializeParallelComm, 25
initializeSubset, 25
interpolateToFaces, 26
meshSubsetPtr_, 30
meshWritten_, 30
nGlobalFaces_, 30
oldInternalFaceIndices_, 30
outputDir_, 30
outsideCellMapPtr_, 30
publicKeyBase64_, 31
read, 26
remoteCellIndices_, 31
remoteCellProcs_, 31
remoteFaceIndices_, 31
spaceTimeWindowExtract, 24
subsetInitialized_, 31
TypeName, 26
updateMetadataTimesteps, 26
useDeltaVarint_, 31
useEncryption_, 32
write, 27
writeBoundaryData, 27
writeCompression_, 32
writeField, 27
writeFoamFileHeader, 27
writeFormat_, 32
writeInitialFields, 27
writeSubsetMesh, 28
Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToMaster< Foam::scalar >
spaceTimeWindowExtract.C, 38
Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToMaster< Foam::vector >
spaceTimeWindowExtract.C, 38
Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces< Foam::scalar >
spaceTimeWindowExtract.C, 38
Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces< Foam::vector >
spaceTimeWindowExtract.C, 38
Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::scalar >
spaceTimeWindowExtract.C, 38
Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::vector >
spaceTimeWindowExtract.C, 38
Foam::sodiumCrypto, 18
available, 19
decryptFromFile, 19
derivePublicKey, 19
derivePublicKeyBase64, 19
encryptToFile, 19, 20
fileExtension, 21
fromBase64, 20
generateKeypair, 20
HEADER_SIZE, 21
initialize, 20
isEncryptedFile, 20
MAGIC, 21
PRIVATE_KEY_SIZE, 22
PUBLIC_KEY_SIZE, 22
readPrivateKeyFromStdin, 21
toBase64, 21
Foam::spaceTimeWindowFvPatchField
autoMap, 34
clone, 34
fixesValue, 35
rmap, 35
spaceTimeWindowFvPatchField, 33, 34
TypeName, 35
updateCoeffs, 35
write, 35
Foam::spaceTimeWindowFvPatchField< Type >, 32
fromBase64
Foam::sodiumCrypto, 20

gatherAndWriteInitialFields
Foam::functionObjects::spaceTimeWindowExtract, 24
gatherAndWriteSubsetMesh
Foam::functionObjects::spaceTimeWindowExtract, 24
gatherFaceDataToMaster
Foam::functionObjects::spaceTimeWindowExtract, 24
gatherFieldToMaster
Foam::functionObjects::spaceTimeWindowExtract, 25
generateKeypair
Foam::sodiumCrypto, 20
globalFaceIndexPtr_
Foam::functionObjects::spaceTimeWindowExtract, 29

HEADER_SIZE
Foam::sodiumCrypto, 21
hasProcessorBoundaryFaces_
Foam::functionObjects::spaceTimeWindowExtract, 29

identifyProcessorBoundaryFaces
Foam::functionObjects::spaceTimeWindowExtract, 25
initialize
Foam::sodiumCrypto, 20
initializeParallelComm
Foam::functionObjects::spaceTimeWindowExtract, 25
initializeSubset
Foam::functionObjects::spaceTimeWindowExtract, 25
interpolateToFaces
Foam::functionObjects::spaceTimeWindowExtract, 26
isDeltaVarintBuffer
Foam::deltaVarintCodec, 16
isDeltaVarintFile

Foam::deltaVarintCodec, 17
isEncryptedFile
Foam::sodiumCrypto, 20

MAGIC
Foam::sodiumCrypto, 21
magic
Foam::deltaVarintCodec, 17
main
spaceTimeWindowInitCase.C, 41
spaceTimeWindowKeygen.C, 43
makePatchFields
Foam, 14
makePatchTypeFieldTypedefs
Foam, 14
meshSubsetPtr_
Foam::functionObjects::spaceTimeWindowExtract, 30
meshWritten_
Foam::functionObjects::spaceTimeWindowExtract, 30

nGlobalFaces_
Foam::functionObjects::spaceTimeWindowExtract, 30

oldInternalFaceIndices_
Foam::functionObjects::spaceTimeWindowExtract, 30
outputDir_
Foam::functionObjects::spaceTimeWindowExtract, 30
outsideCellMapPtr_
Foam::functionObjects::spaceTimeWindowExtract, 30

PRIVATE_KEY_SIZE
Foam::sodiumCrypto, 22
PUBLIC_KEY_SIZE
Foam::sodiumCrypto, 22
publicKeyBase64_
Foam::functionObjects::spaceTimeWindowExtract, 31

README.md, 36
read
Foam::functionObjects::spaceTimeWindowExtract, 26
readPrivateKeyFromStdin
Foam::sodiumCrypto, 21
readScalar
Foam::deltaVarintCodec, 17
readVector
Foam::deltaVarintCodec, 17
remoteCellIndices_
Foam::functionObjects::spaceTimeWindowExtract, 31
remoteCellProcs_
Foam::functionObjects::spaceTimeWindowExtract, 31
remoteFaceIndices_
Foam::functionObjects::spaceTimeWindowExtract, 31
rmap
Foam::spaceTimeWindowFvPatchField, 35

sodiumCrypto.C, 36
sodiumCrypto.H, 36
spaceTimeWindowExtract
Foam::functionObjects::spaceTimeWindowExtract, 24
spaceTimeWindowExtract.C, 37
Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToMaster< Foam::scalar >, 38
Foam::functionObjects::spaceTimeWindowExtract::gatherFieldToMaster< Foam::vector >, 38
Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces< Foam::scalar >, 38
Foam::functionObjects::spaceTimeWindowExtract::interpolateToFaces< Foam::vector >, 38
Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::scalar >, 38
Foam::functionObjects::spaceTimeWindowExtract::writeField< Foam::vector >, 38
spaceTimeWindowExtract.H, 39
spaceTimeWindowFvPatchField
Foam::spaceTimeWindowFvPatchField, 33, 34
spaceTimeWindowFvPatchField.C, 39
spaceTimeWindowFvPatchField.H, 39
spaceTimeWindowFvPatchFields.C, 40
spaceTimeWindowFvPatchFields.H, 40
spaceTimeWindowInitCase.C, 40
copyDirectory, 41
copyFile, 41
decryptBoundaryFile, 41
main, 41
writeControlDict, 42
writeInitialField, 42
spaceTimeWindowKeygen.C, 42
main, 43
subsetInitialized_
Foam::functionObjects::spaceTimeWindowExtract, 31

toBase64
Foam::sodiumCrypto, 21
TypeName
Foam::functionObjects::spaceTimeWindowExtract, 26
Foam::spaceTimeWindowFvPatchField, 35

updateCoeffs
Foam::spaceTimeWindowFvPatchField, 35
updateMetadataTimesteps

Foam::functionObjects::spaceTimeWindowExtract, 26
useDeltaVarint_
Foam::functionObjects::spaceTimeWindowExtract, 31
useEncryption_
Foam::functionObjects::spaceTimeWindowExtract, 32

write
Foam::deltaVarintCodec, 17
Foam::functionObjects::spaceTimeWindowExtract, 27
Foam::spaceTimeWindowFvPatchField, 35
writeBoundaryData
Foam::functionObjects::spaceTimeWindowExtract, 27
writeCompression_
Foam::functionObjects::spaceTimeWindowExtract, 32
writeControlDict
spaceTimeWindowInitCase.C, 42
writeField
Foam::functionObjects::spaceTimeWindowExtract, 27
writeFoamFileHeader
Foam::functionObjects::spaceTimeWindowExtract, 27
writeFormat_
Foam::functionObjects::spaceTimeWindowExtract, 32
writeInitialField
spaceTimeWindowInitCase.C, 42
writeInitialFields
Foam::functionObjects::spaceTimeWindowExtract, 27
writeSubsetMesh
Foam::functionObjects::spaceTimeWindowExtract, 28