Getting Started with ITK

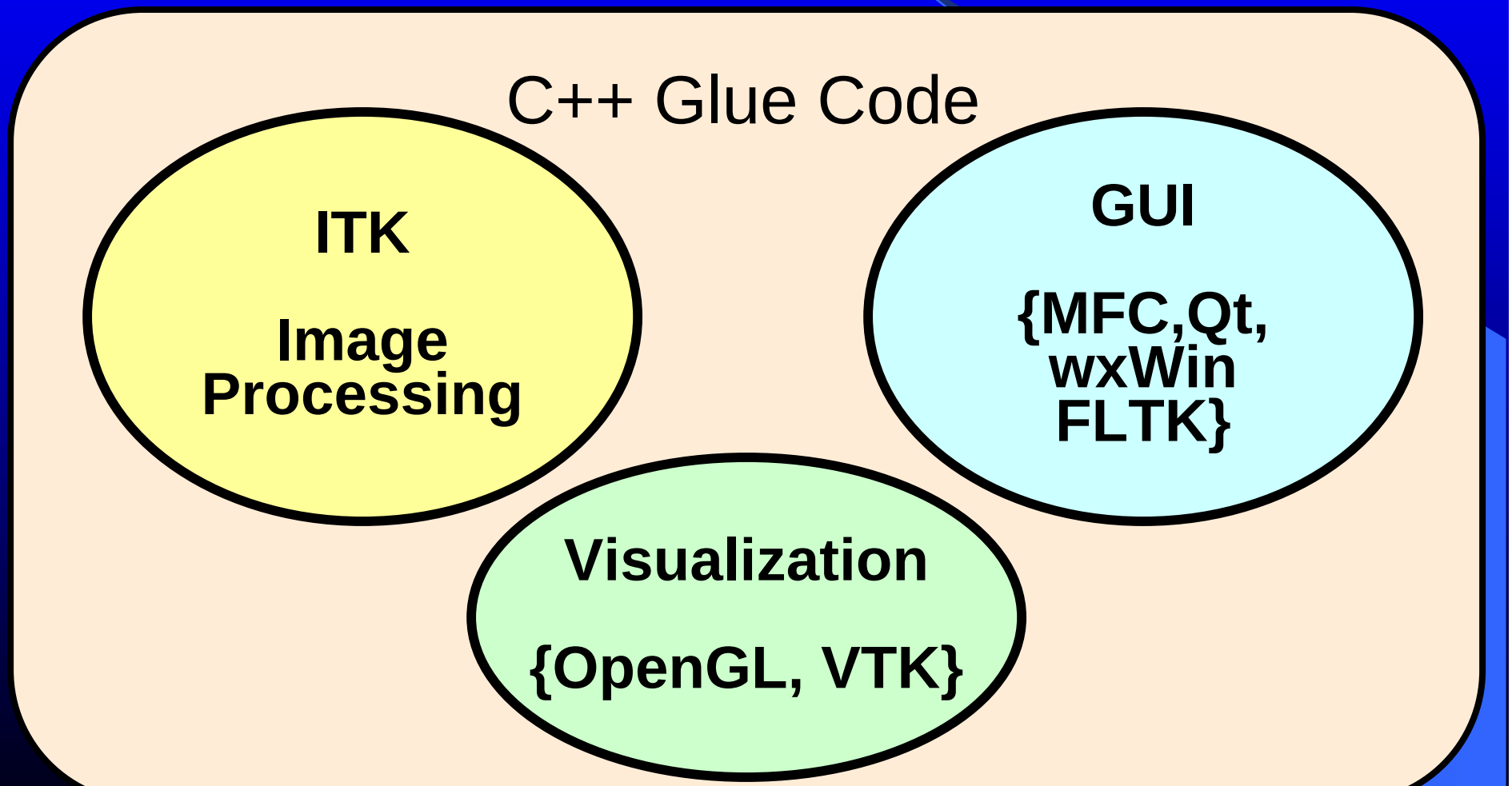
(Instructions for Current Release)

Luis Ibáñez
Will Schroeder
Brad King
Insight Software Consortium

What is ITK

- Image Processing
- Segmentation
- Registration
- No Graphical User Interface (GUI)
- No Visualization

How to Integrate ITK in you application



What do I need ?

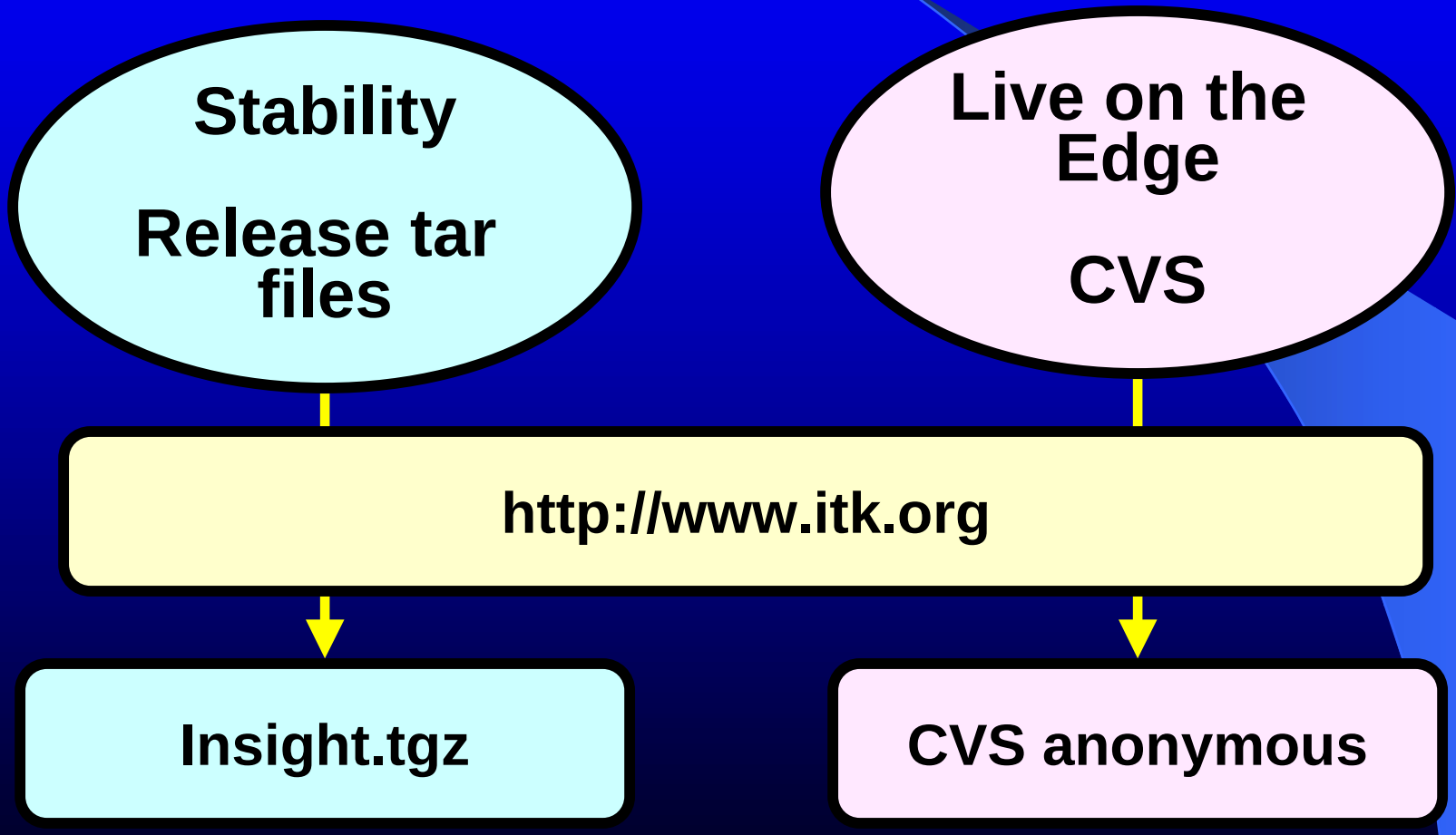
C++ Compiler

gcc 2.95 – 3.1
Visual C++ 6.0
Visual .NET
Intel 5.0
IRIX CC
Borland 5.0
Mac - gcc

CMake

www.cmake.org

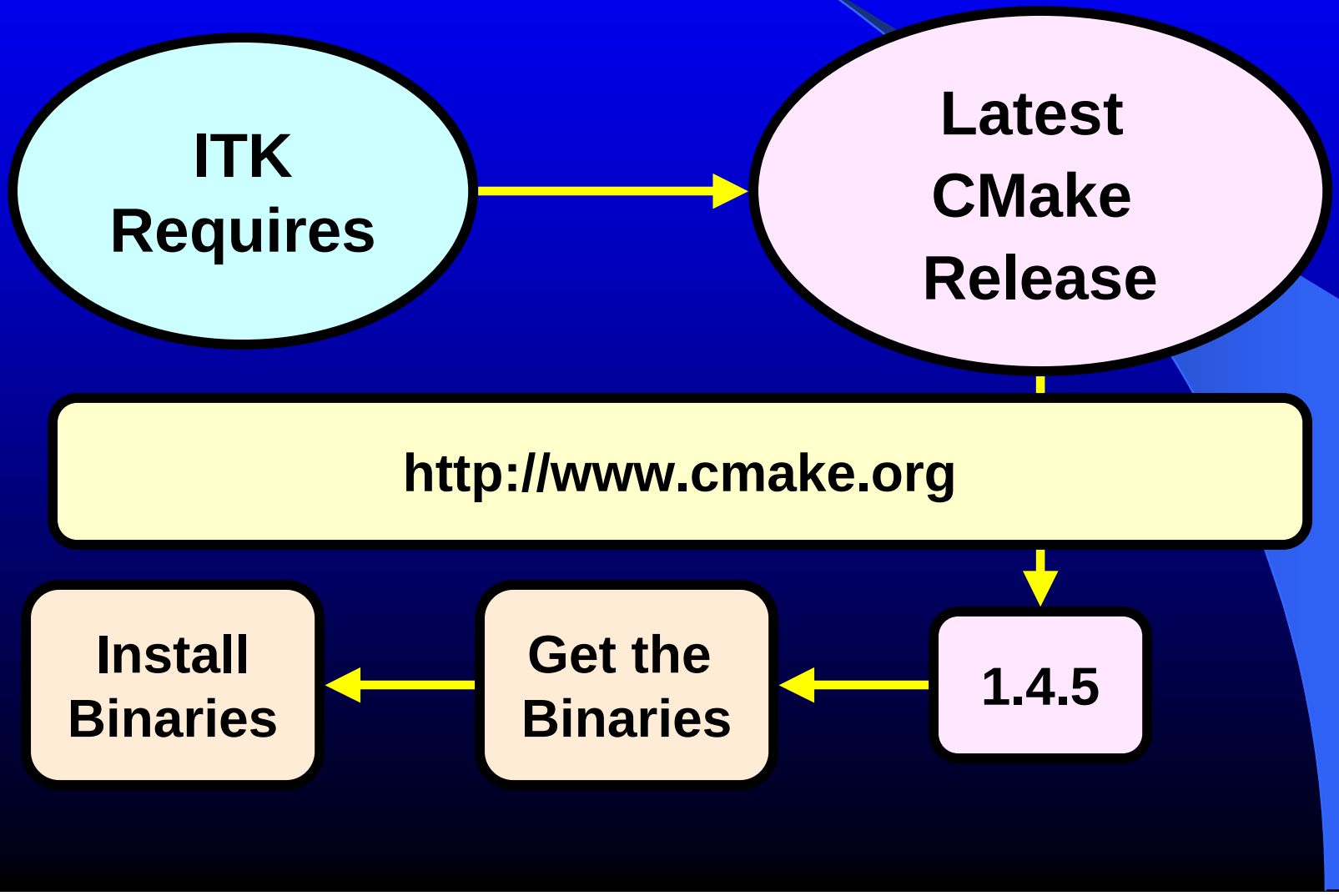
Step 1. Download ITK



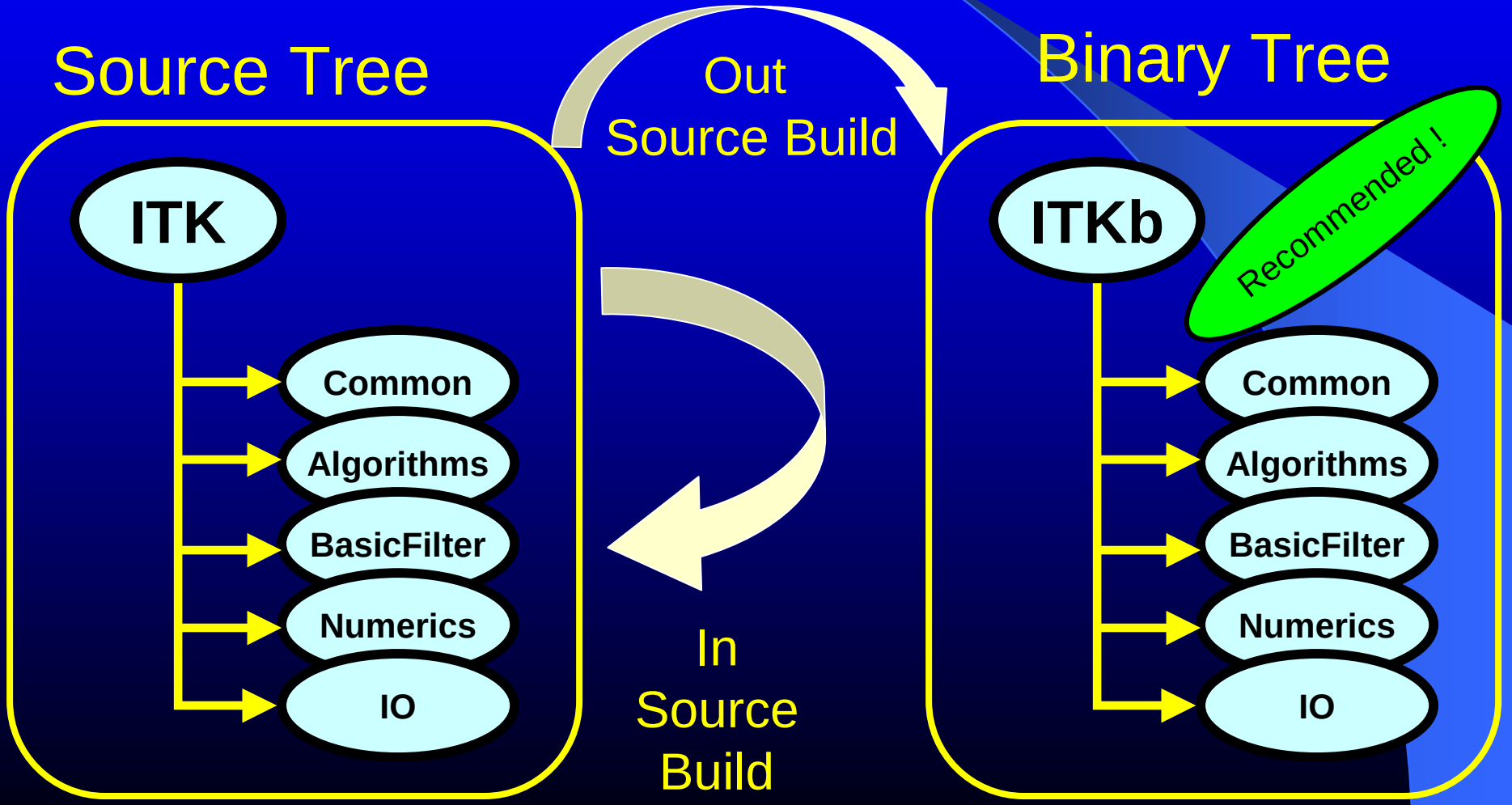
Step 1: Download ITK

- Get packaged releases:
  - http://www.itk.org/HTML/Download.php
- Download from CVS (development version)
  - cvs -d :pserver:anonymous@www.itk.org:/cvsroot/Insight login
  - (respond with password "insight")
  - cvs -d :pserver:anonymous@www.itk.org:/cvsroot/Insight co Insight

Step 2. Download CMake



Step 3. Configure ITK

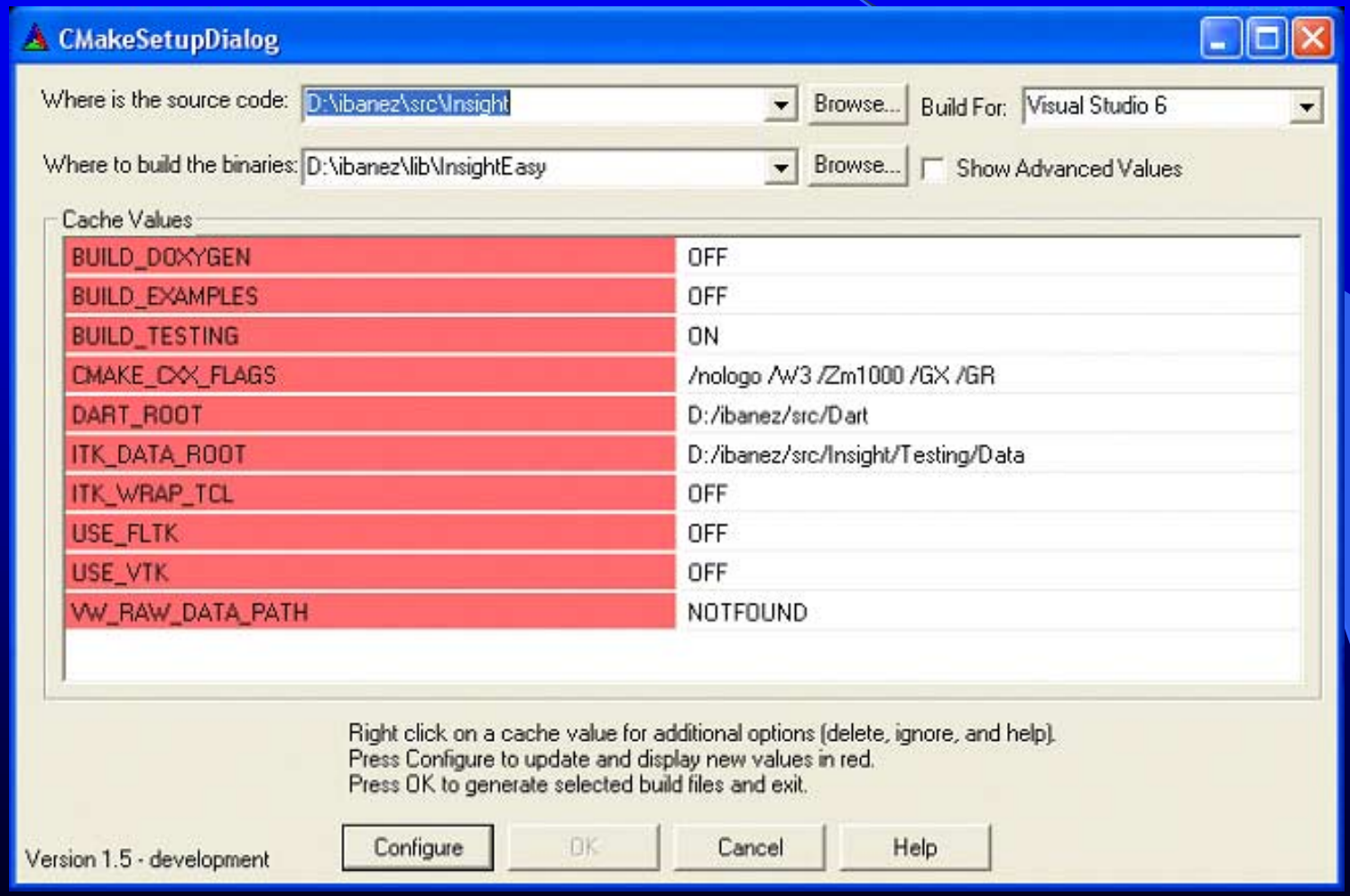


Step 3. Configure - Easy Start

- Run CMake
- Select the SOURCE directory
- Select the BINARY directory
- Select the COMPILER
- Select CONFIGURE and then OK buttons

(Note: CMake works iteratively. As new options are enabled, new CMake variables show up in red. Continue to select CONFIGURE until no red appears (no changes) then finally select OK to produce workspaces / makefiles)

Step 3. Configure - Easy Start



Step 3. Configure - Easy Start

- Disable BUILD_DOXYGEN
- Disable BUILD_EXAMPLES
- Enable BUILD_TESTING
- Disable USE_FLTK
- Disable USE_VTK
- Disable ITK_WRAP_TCL

(Note: BUILD_TESTING can be disabled; this will speed the build process but produce no executables (as referred to in Step 5 in the next slides))

Step 3. Configure - Easy Start

- Ignore CMAKE_CXX_FLAGS
- Ignore DART_ROOT
- Ignore ITK_DATA_ROOT
- Ignore VW_RAW_DATA_PATH

Step 4. Build Project

- Open ITK.dsw in the BINARY Directory (assuming MSVC compiler)
- Select ALL_BUILD project
  - Select configuration
    - Debug *(recommended initially)*
    - Release
    - RelWithDebugInfo
    - MinSizeRel
- Build…it will take about 1 hour …
  - *(if BUILD_TESTING is disabled, less than 15 minutes)*

Step 4. Build Project

- Most of ITK classes are C++ Templates
- Basic libraries are small—they only contain non-templated classes
- Basic libraries are built in about 15 min

Step 5. Verify the Build

- Libraries and test Executables will be found in

  ITK_BINARY / bin / { Debug, Release }

  - *The actual location depends on the configuration chosen in MSVC compiler*
  - *(Executables will be present only if BUILD_TESTING was enabled in CMake)*

Step 5. Verify the Build

The following libraries should be found

- ITKCommon
- ITKBasicFilters
- ITKAlgorithms
- ITKNumerics
- ITKFEM
- ITKIO
- ITKStatistics
- VXLNumerics
- itkpng
- itkzlib
- ITKMetaIO

Step 5. Verify the Build

The following executables should be found

- itkCommonTests
- itkBasicFiltersTests
- itkAlgorithmsTests
- itkNumericsTests
- itkIOTests

Step 5. Verify the Build

The following executables should be found

- itkSpatialObjectTests
- itkFEMTests
- itkStatisticsTests
- vnlTests

Step 5. Verify the Build

The following executables should be found

- itkCommonHeaderTest
- itkBasicFiltersHeaderTest
- itkAlgorithmsHeaderTest
- itkNumericsHeaderTest
- itkIOHeaderTest
- itkSpatialObjectHeaderTest

Step 5. Verify the Build

- Run ONE of the tests
- Murphy's Law guarantees that if there is only ONE test failing, it will be the one you randomly select !

Step 5. Verify the Build

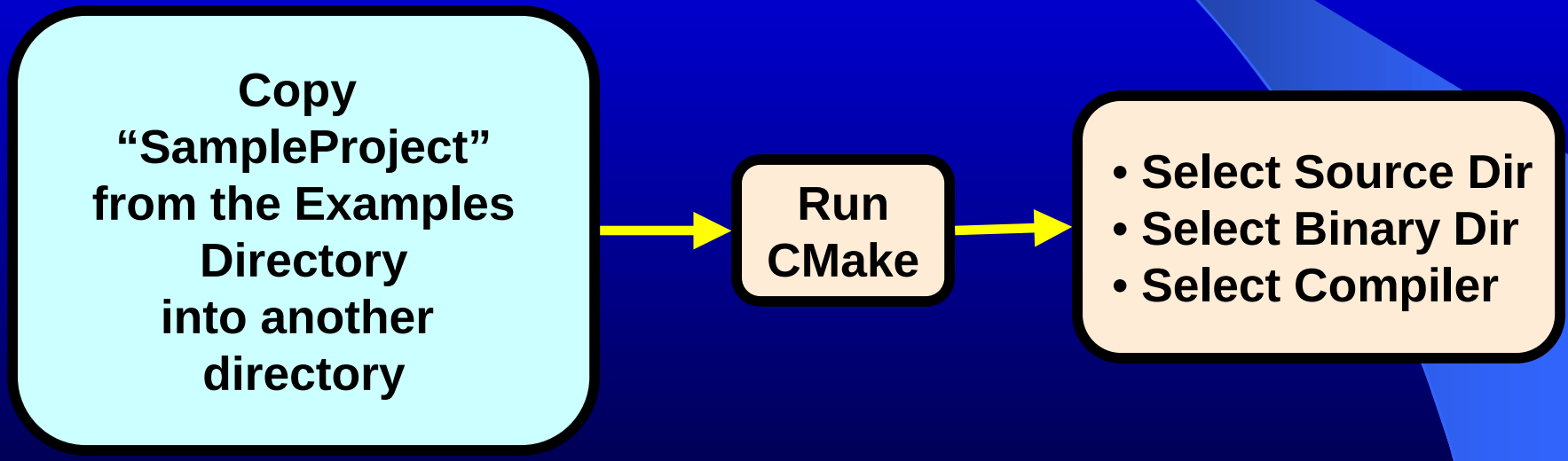
- *Assuming that BUILD_TESTING was enabled:*
  - The test organization reflects the source tree structure
  - For example, in order to test the GradientImageFilter execute

    itkBasicFiltersTests.exe itkGradientImageFilter
  - Preferred way to execute tests:
    - cd into BINARY directory (as set in CMake)
    - ctest –R itkGradientImageFilter

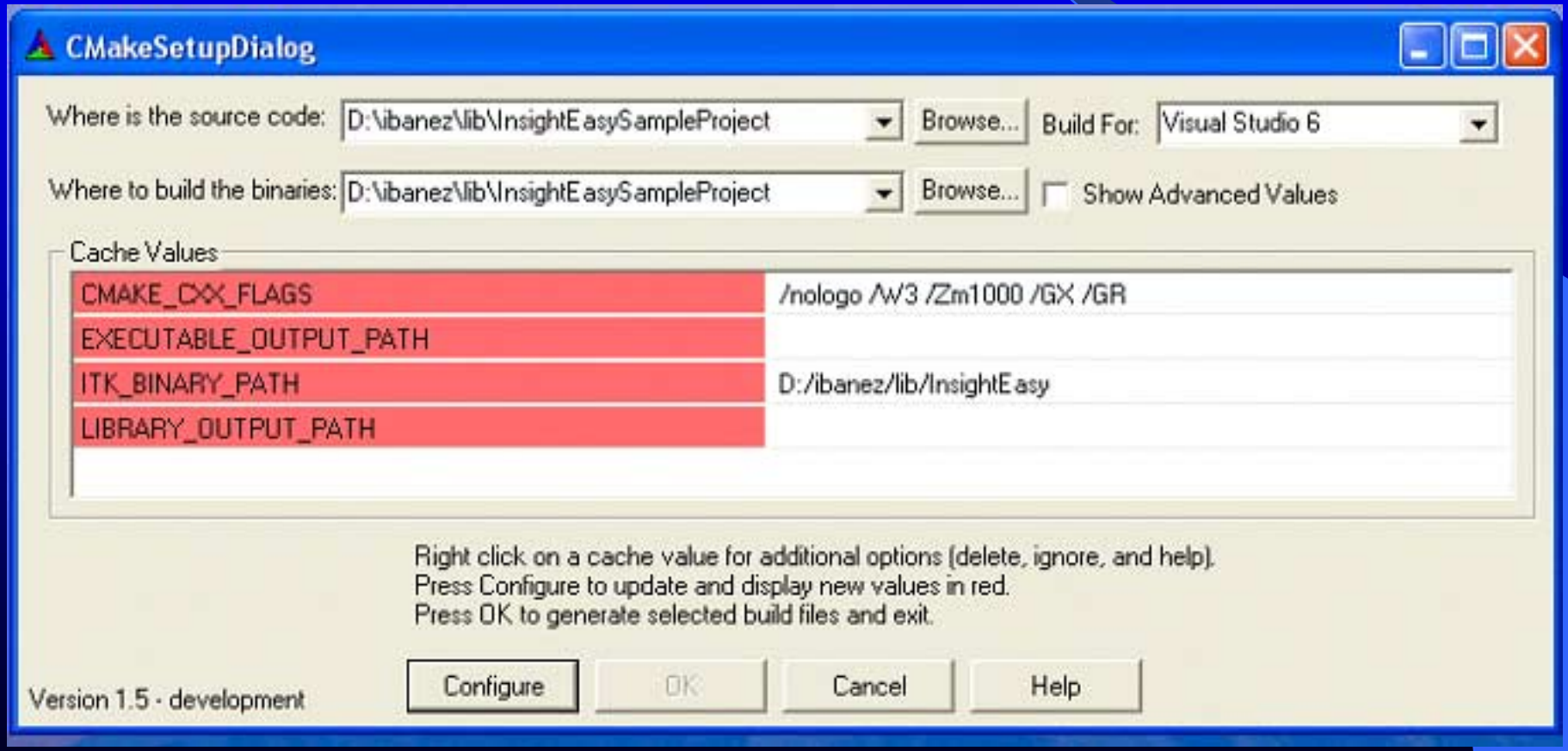
      (-R says any test matching this string is executed;

      without –R all tests are executed)
    - ctest is a companion program to CMake

Step 6. Use ITK from an external Project



Step 6. Use ITK from an external Project



Step 6. Use ITK from an external Project

- Ignore CMAKE_CXX_FLAGS
- Ignore EXECUTABLE_OUTPUT_PATH
- Ignore LIBRARY_OUTPUT_PATH
- Point ITK_BINARY_PATH to the binary directory where ITK was built

Step 7. Build Sample Project

- Open SampleProject.dsw generated by CMake
- Select ALL_BUILD project
- Build it
  …It will take about 10 seconds …

Step 8. Run the example

- Locate the file itkSampleProject.exe
- Run it…
- It should produce the message:

  Test Passed !

Step 9. Start your own project

- Create a clean new directory
- Write a CMakeLists.txt file
- Writer a simple .cxx file
- Configure with CMake
- Build
- Run

Step 10. Writing CMakeLists.txt

```
PROJECT( myProject )

INCLUDE (${CMAKE_ROOT}/Modules/FindITK.cmake)
IF ( USE_ITK_FILE )
        INCLUDE(${USE_ITK_FILE})
ENDIF( USE_ITK_FILE )

ADD_EXECUTABLE( myProject    myProject.cxx )

TARGET_LINK_LIBRARIES ( myProject  ${ITK_LIBRARIES} )
```

Step 11. Writing myProject.cxx

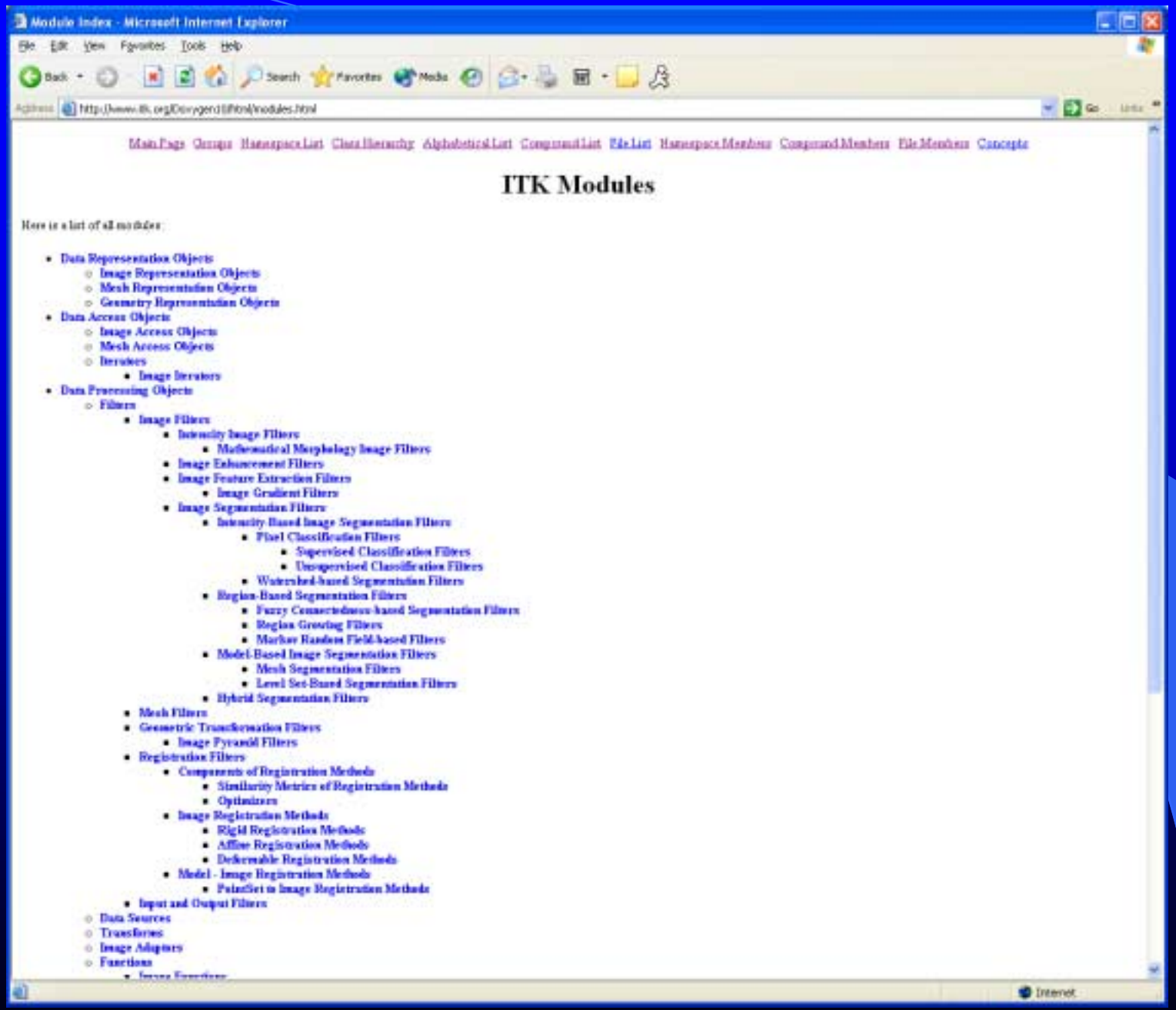
```
#include "itkImage.h"
#include "itkImageFileReader.h"
#include "itkGradientMagnitudeImageFilter.h"

int main( int argc, char **argv ) {
  typedef itk::Image<unsigned short,2>          ImageType;
  typedef itk::ImageFileReader<ImageType>       ReaderType;
  typedef itk::GradientMagnitudeImageFilter<
                        ImageType,ImageType>    FilterType;

  ReaderType::Pointer reader = ReaderType::New();
  FilterType::Pointer filter = FilterType::New();

  reader->SetFileName( argv[1] );
  filter->SetInput( reader->GetOutput() );
  filter->Update();
  return 0;
}
```

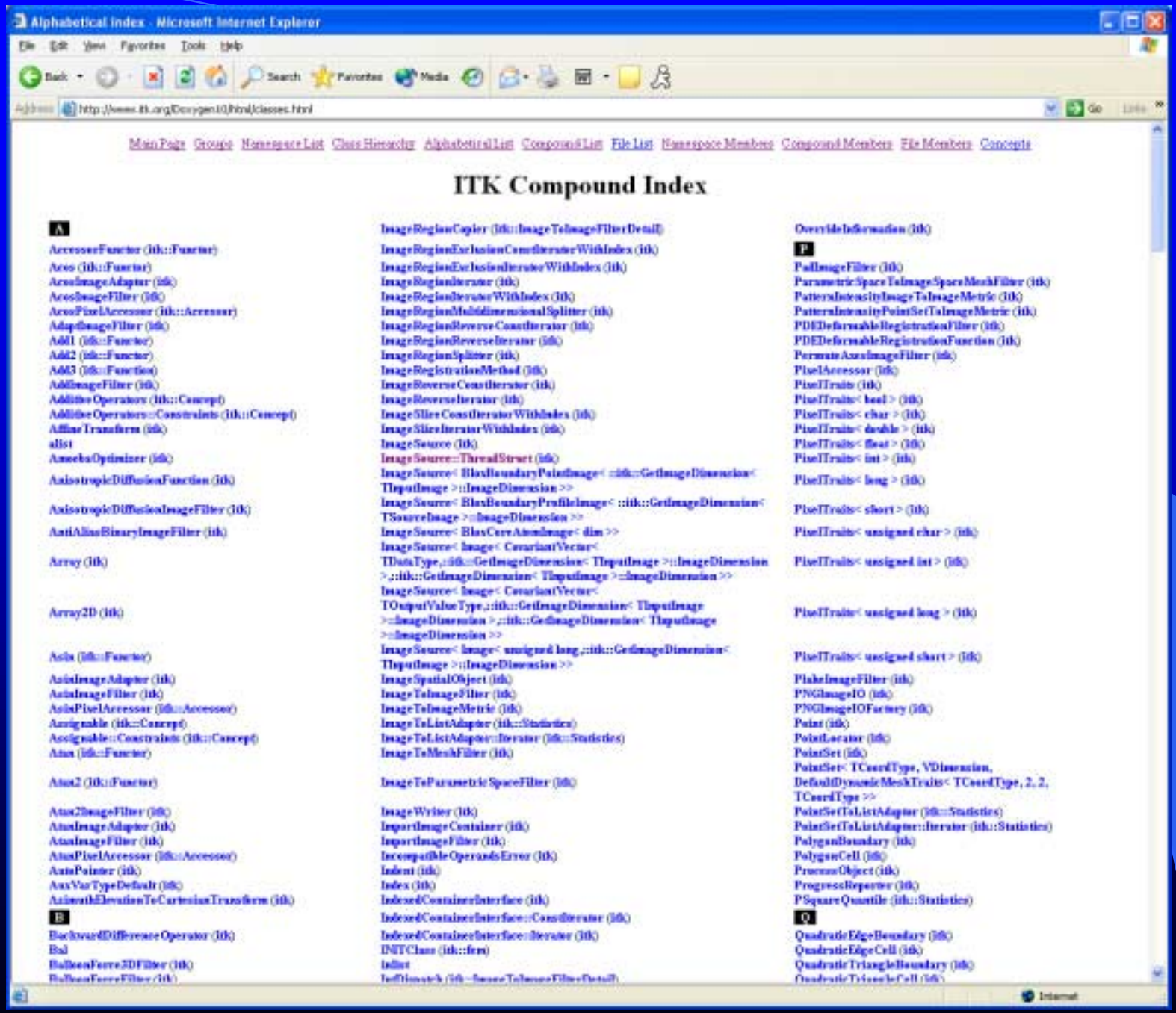
Step 12. How to find what you need

http://www.itk.org/Doxygen10/html/index.html

- Follow the link Alphabetical List
- Follow the link Groups
- Post to the insight-users mailing list


Insight Toolkit - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://www.itk.org/Doxygen10/html/index.html
Go
Main Page Groups Namespace List Class Hierarchy Alphabetical List Compound List File List Namespace Members Compound Members File Members Concepts
Insight Toolkit
1.0.0
itk
Introduction
Welcome to the National Library of Medicine Insight Segmentation and Registration Toolkit (ITK). ITK is an open-source software system to support the Visible Human Project. Currently under active development, ITK employs leading edge segmentation and registration algorithms in two, three and more dimensions.
Home Page
The Home Page of the Insight toolkit can be found at:
http://www.itk.org
How to use this documentation
This documentation describes the API of the Toolkit. The Modules link presents a hierarchy of classes organized according to their functionality. The Related Pages link presents design details, in particular the use of the data pipeline model and the philosophy of itkstore.
Generated at Wed Oct 16 01:12:17 2002 for ITK by
1.2.15 written by Dimitri van Heesch, © 1997-2000
Internet


Module Index - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
http://www.itk.org/Doxygen118html/modules.html
Main Page Groups Namespace List Class Hierarchy Alphabetical List Compound List File List Namespace Members Compound Members File Members Concepts
ITK Modules
Here is a list of all modules:
Data Representation Objects
Image Representation Objects
Mesh Representation Objects
Geometry Representation Objects
Data Access Objects
Image Access Objects
Mesh Access Objects
Iterators
Image Iterators
Data Processing Objects
Filters
Image Filters
Intensity Image Filters
Mathematical Morphology Image Filters
Image Enhancement Filters
Image Feature Extraction Filters
Image Gradient Filters
Image Segmentation Filters
Intensity Based Image Segmentation Filters
Pixel Classification Filters
Supervised Classification Filters
Unsupervised Classification Filters
Watershed-based Segmentation Filters
Region-Based Segmentation Filters
Fuzzy Connectedness-based Segmentation Filters
Region Growing Filters
Markov Random Field-based Filters
Model-Based Image Segmentation Filters
Mesh Segmentation Filters
Level Set-Based Segmentation Filters
Hybrid Segmentation Filters
Mesh Filters
Geometric Transformation Filters
Image Pyramid Filters
Registration Filters
Components of Registration Methods
Similarity Metrics of Registration Methods
Optimizers
Image Registration Methods
Rigid Registration Methods
Affine Registration Methods
Deformable Registration Methods
Model - Image Registration Methods
PointSet to Image Registration Methods
Input and Output Filters
Data Sources
Transforms
Image Adaptors
Functions
Internet


Alphabetical Index - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://www.itk.org/Doxygen10/html/classes.html
Main Page Groups Namespace List Class Hierarchy Alphabetical List Compound List File List Namespace Members Compound Members File Members Concepts
ITK Compound Index
A
AccessorFunctor (itk::Functor)
Acos (itk::Functor)
AcosImageAdaptor (itk)
AcosImageFilter (itk)
AcosPixelAccessor (itk::Accessor)
AdaptImageFilter (itk)
Add1 (itk::Functor)
Add2 (itk::Functor)
Add3 (itk::Functor)
AddImageFilter (itk)
AdditiveOperators (itk::Concept)
AdditiveOperators::Constraints (itk::Concept)
AffineTransform (itk)
alist
AmoebaOptimizer (itk)
AnisotropicDiffusionFunction (itk)
AnisotropicDiffusionImageFilter (itk)
AntiAliasBinaryImageFilter (itk)
Array (itk)
Array2D (itk)
Asin (itk::Functor)
AsinImageAdaptor (itk)
AsinImageFilter (itk)
AsinPixelAccessor (itk::Accessor)
Assignable (itk::Concept)
Assignable::Constraints (itk::Concept)
Atan (itk::Functor)
Atan2 (itk::Functor)
Atan2ImageFilter (itk)
AtanImageAdaptor (itk)
AtanImageFilter (itk)
AtanPixelAccessor (itk::Accessor)
AutoPointer (itk)
AutoVarTypeDefault (itk)
AzimuthElevationToCartesianTransform (itk)
B
BackwardDifferenceOperator (itk)
Bal
BalloonForce3DFilter (itk)
BalloonForceFilter (itk)
ImageRegionCopier (itk::ImageToImageFilterDetail)
ImageRegionExclusionConstIteratorWithIndex (itk)
ImageRegionExclusionIteratorWithIndex (itk)
ImageRegionIterator (itk)
ImageRegionIteratorWithIndex (itk)
ImageRegionMultidimensionalSplitter (itk)
ImageRegionReverseConstIterator (itk)
ImageRegionReverseIterator (itk)
ImageRegionSplitter (itk)
ImageRegistrationMethod (itk)
ImageReverseConstIterator (itk)
ImageReverseIterator (itk)
ImageSliceConstIteratorWithIndex (itk)
ImageSliceIteratorWithIndex (itk)
ImageSource (itk)
ImageSource::ThreadStruct (itk)
ImageSource< BlurBoundaryPointImage< ::itk::GetImageDimension< TInputImage >::ImageDimension >>
ImageSource< BlurBoundaryProfileImage< ::itk::GetImageDimension< TSourceImage >::ImageDimension >>
ImageSource< BlurCurvAtomImage< dim >>
ImageSource< Image< CovariantVector< TDataType,::itk::GetImageDimension< TInputImage >::ImageDimension >,::itk::GetImageDimension< TInputImage >::ImageDimension >>
ImageSource< Image< CovariantVector< TOutputValueType,::itk::GetImageDimension< TInputImage >::ImageDimension >,::itk::GetImageDimension< TInputImage >::ImageDimension >>
ImageSource< Image< unsigned long,::itk::GetImageDimension< TInputImage >::ImageDimension >>
ImageSpatialObject (itk)
ImageToImageFilter (itk)
ImageToImageMetric (itk)
ImageToListAdaptor (itk::Statistics)
ImageToListAdaptor::Iterator (itk::Statistics)
ImageToMeshFilter (itk)
ImageToParametricSpaceFilter (itk)
ImageWriter (itk)
ImportImageContainer (itk)
ImportImageFilter (itk)
IncompatibleOperandsError (itk)
Indent (itk)
Index (itk)
IndexedContainerInterface (itk)
IndexedContainerInterface::ConstIterator (itk)
IndexedContainerInterface::Iterator (itk)
INITClass (itk::fem)
InList
OverrideInformation (itk)
P
PadImageFilter (itk)
ParametricSpaceToImageSpaceMeshFilter (itk)
PatternIntensityImageToImageMetric (itk)
PatternIntensityPointSetToImageMetric (itk)
PDEDeformableRegistrationFilter (itk)
PDEDeformableRegistrationFunction (itk)
PermuteAxesImageFilter (itk)
PixelAccessor (itk)
PixelTraits (itk)
PixelTraits< bool > (itk)
PixelTraits< char > (itk)
PixelTraits< double > (itk)
PixelTraits< float > (itk)
PixelTraits< int > (itk)
PixelTraits< long > (itk)
PixelTraits< short > (itk)
PixelTraits< unsigned char > (itk)
PixelTraits< unsigned int > (itk)
PixelTraits< unsigned long > (itk)
PixelTraits< unsigned short > (itk)
PlaneImageFilter (itk)
PNGImageIO (itk)
PNGImageIOFactory (itk)
Point (itk)
PointLocator (itk)
PointSet (itk)
PointSet< TCoordType, VDimension, DefaultDynamicMeshTraits< TCoordType, 2, 2, TCoordType >>
PointSetToListAdaptor (itk::Statistics)
PointSetToListAdaptor::Iterator (itk::Statistics)
PolygonBoundary (itk)
PolygonCell (itk)
ProcessObject (itk)
ProgressReporter (itk)
PSquareQuantile (itk::Statistics)
Q
QuadraticEdgeBoundary (itk)
QuadraticEdgeCell (itk)
QuadraticTriangleBoundary (itk)
Internet

Additional Resources

- User Mailing List
  - http://www.itk.org/mailman/listinfo/insight-users
- Further Documentation
  - InsightDocuments (cvs checkout)

    cvs -d :pserver:anonymous@www.itk.org:/cvsroot/Insight login

    with password "insight"...then get the source/documentation as follows:

    cvs -d :pserver:anonymous@www.itk.org:/cvsroot/Insight checkout

Enjoy ITK !