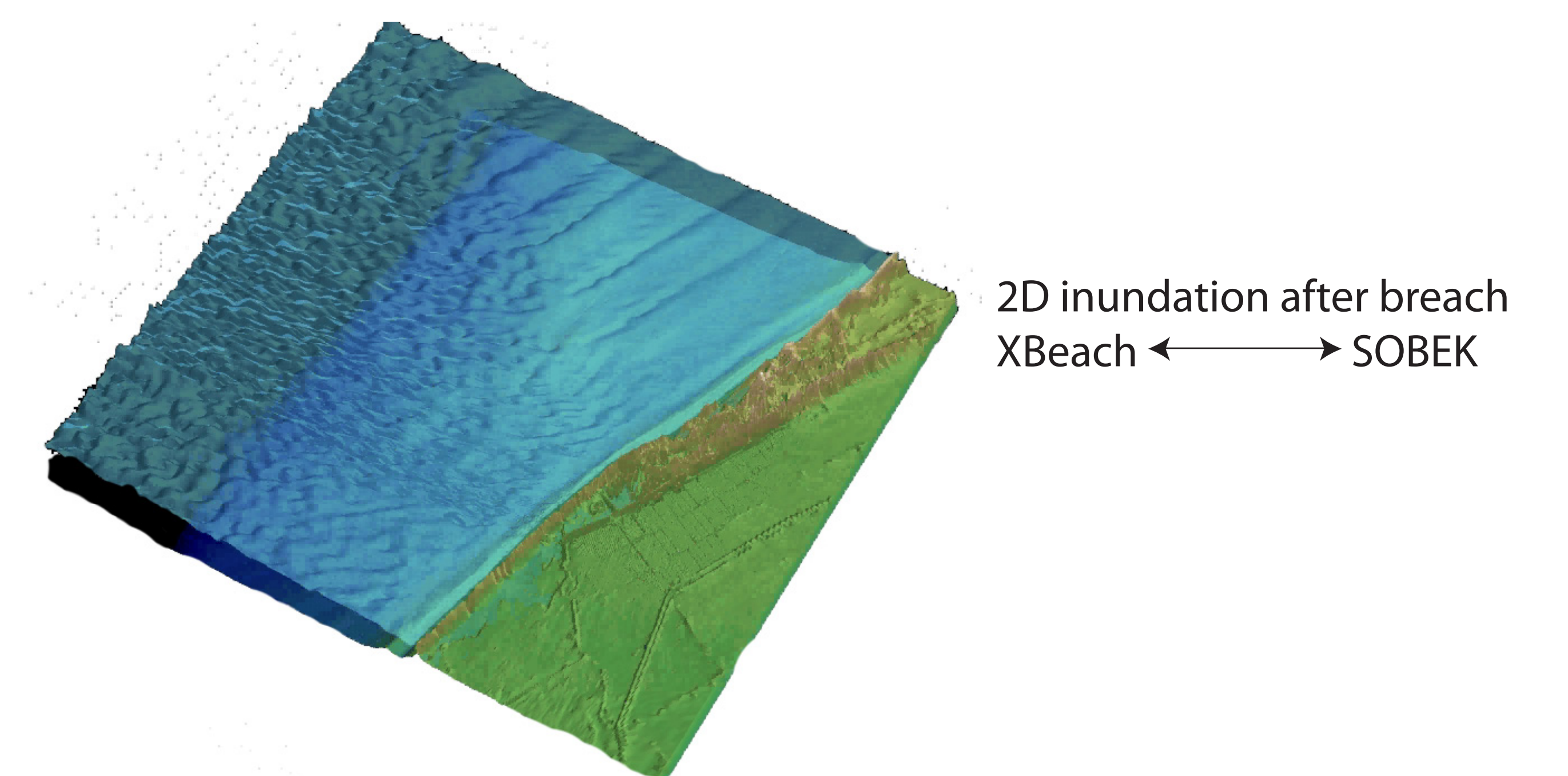
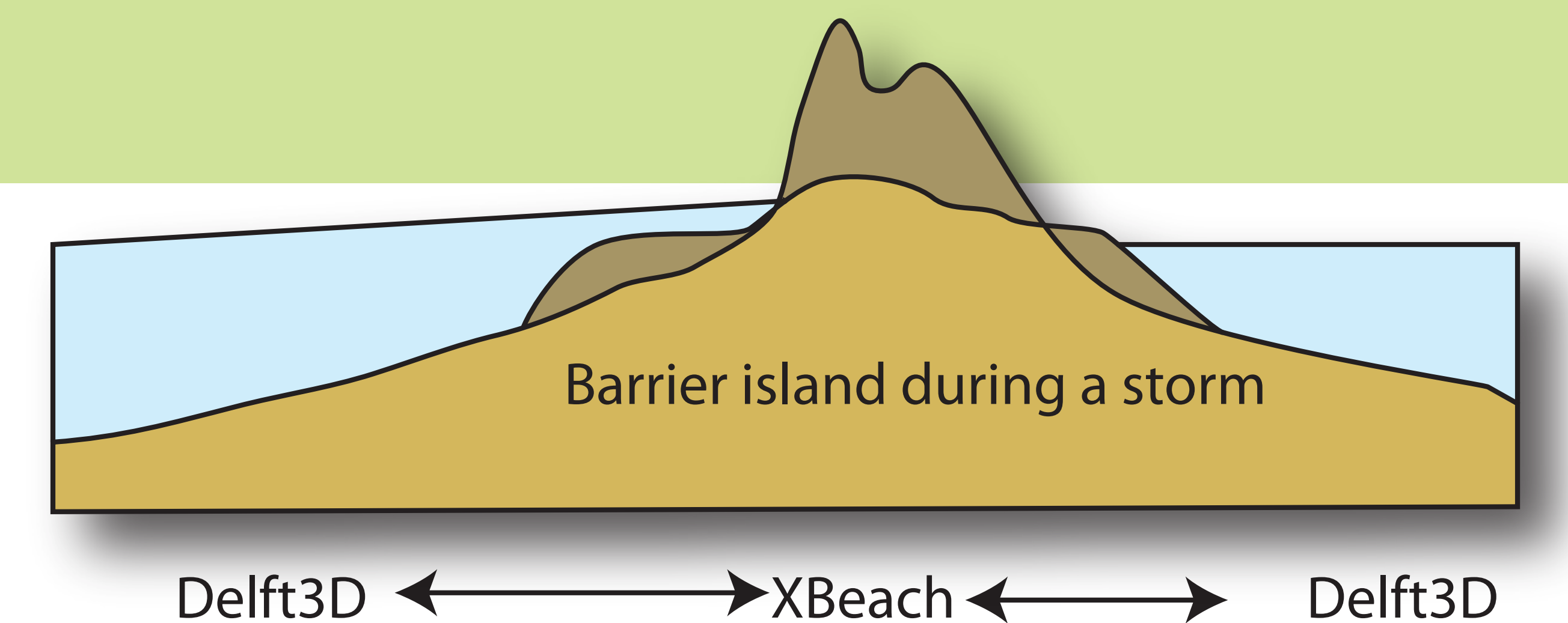
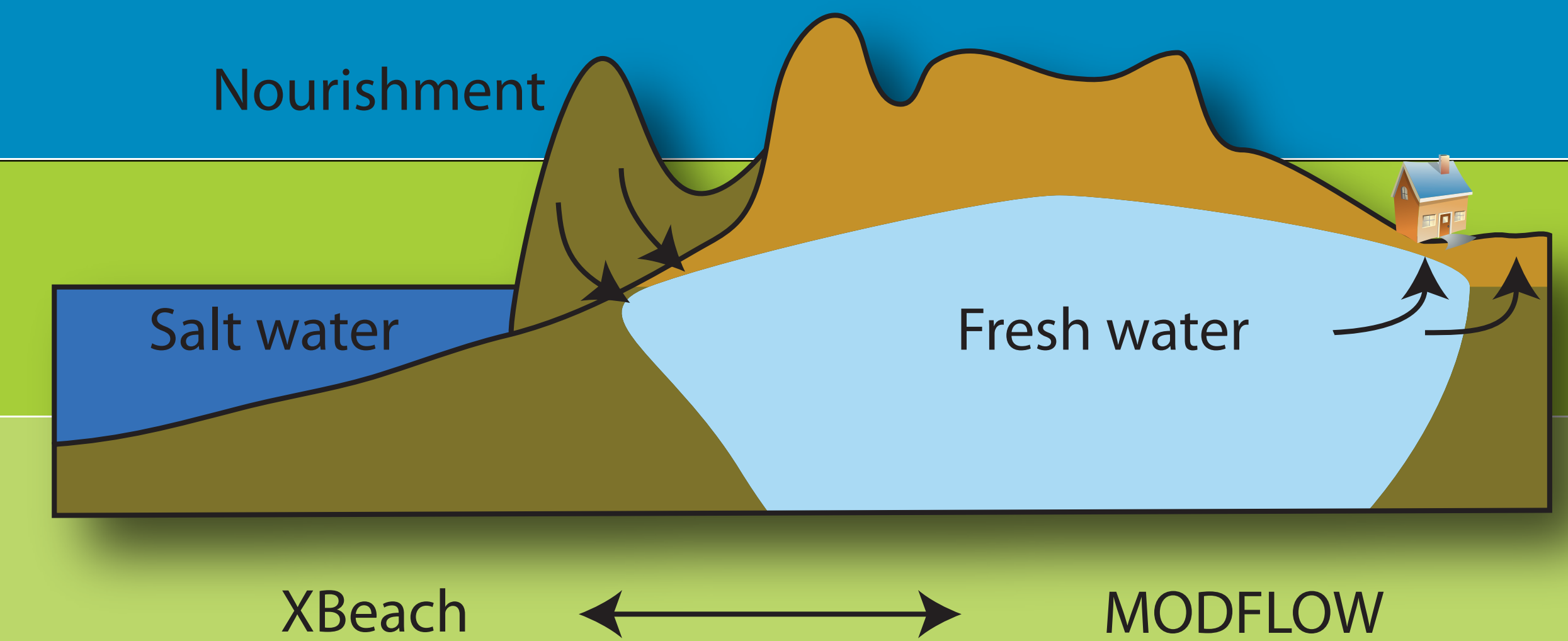


Introduction

We show the coupling of the nearshore erosion model XBeach to two modelling frameworks: ESMF and OpenMI. We compare the two implementations and give suggestions for easy model coupling.

Possible applications of coupled XBeach



Reducing the invasiveness of modelling frameworks

XBeach

A two-dimensional model for wave propagation, long waves and mean flow, sediment transport and morphological changes of the nearshore area, beaches, dunes and backbarrier during storms. It consists of 17000 lines of Fortran90.

From XBeach to libX-Beach

The main XBeach program was rewritten to a library (.so). Making a library resulted in 140 extra lines of code (29 for C compatibility).

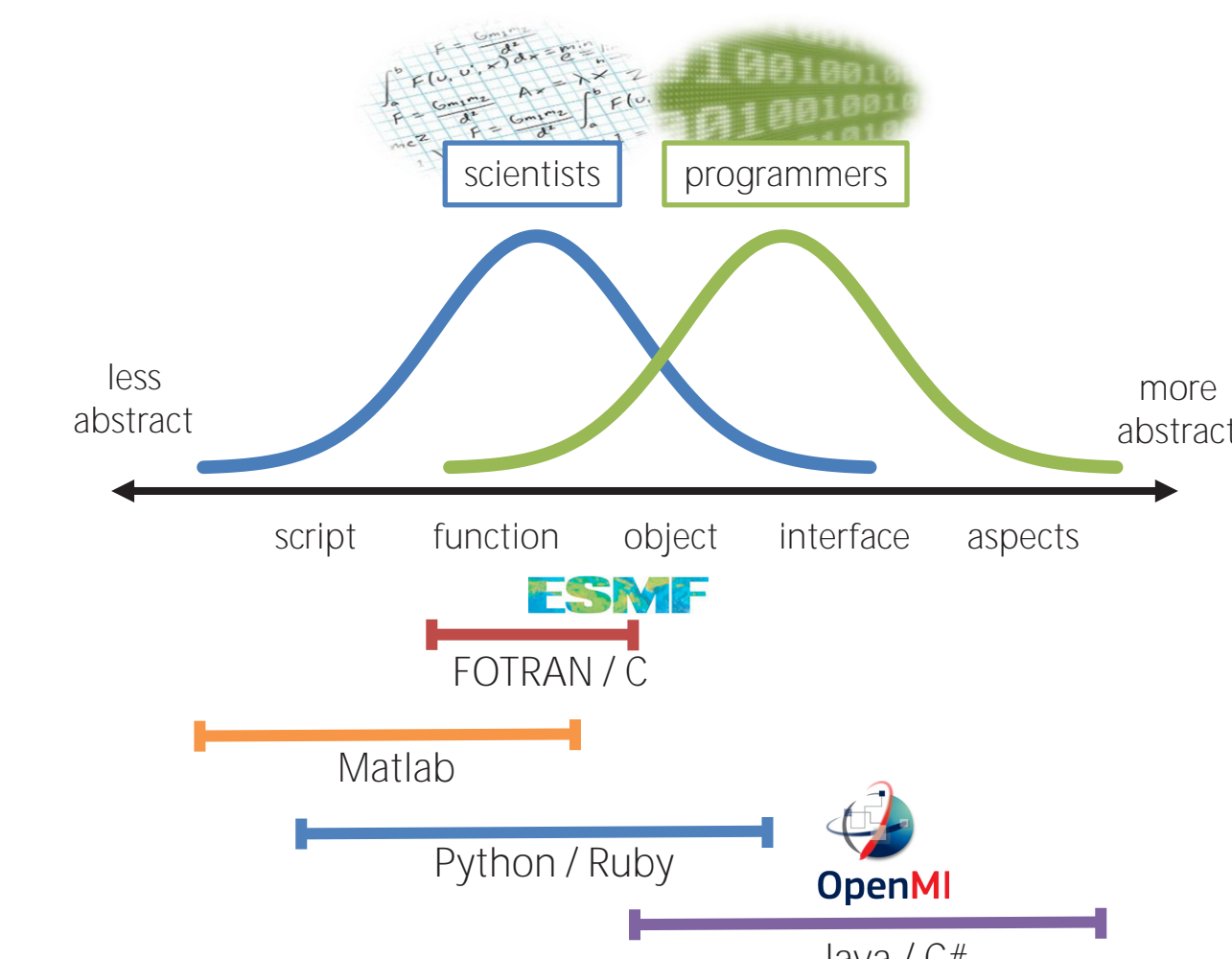
XBeach + OpenMI

For OpenMI a C# wrapper around libx-beach was written (65 lines), the OpenMI specification was implemented in an XBeachComponent (117 lines). Unit-tests were also added (189 lines).

XBeach + ESMF

An XBeachComponent (93 lines) and a coupler were created (44 lines) and tests (194 lines).

Frameworks



ESMF	Description	OpenMI 2.0
ESMF_AppDriver	Main program/controller	OpenMI gui + command line runner
ESMF_GridComp	Component with input and output (run, initialize, finalize)	IBaseLinkableComponent
ESMF_CplComp	Maps input to a output state (used for conversion of units, grids, spatial transformation).	SDK / OpenMI Tools
ESMF_State	Connectable input and output items	IBaseExchangeItem / IBaseInput / IBaseOutput
ESMF_Array	Values (arrays, datatype) + attributes (units, spatial information)	IValueDefinition, IBaseExchangeItem, IElementSet
ESMF_Grid	Geospatial representation of connected items	IElementSet
ESMF_Field	Grid with an array, staggering, location	IElementSet, SDK
ESMF_Time	Time management	ITime
ESMF_Calendar	Time management	ITime
ESMF_TimeInterval	Time management	ITime

ESMF:

```
! Initialize XBeach
rc = init()
call ESMF_GridCompPrint(gcomp, rc=rc)
! Create the grid
rc = getintparameter("nx", nx, 2)
rc = getintparameter("ny", ny, 2)
grid = ESMF_GridCreateShapeTile(maxIndex=(/
nx,ny/), name="xbeachgrid", rc=rc)
stateName = "Coast"
exportState = ESMF_StateCreate(stateName,
stateType=ESMF_STATE_EXPORT, rc=rc)
bundleName = "Bathymetry"
bundle = ESMF_FieldBundleCreate(name=bundleName,
rc=rc)
array = ESMF_ArrayCreate(s%zb, distgrid=distgrid)
field(1) = ESMF_FieldCreate(grid, array=array,
name="zb", rc=rc)
call ESMF_FieldBundleAdd(bundle, field(1), rc)
call ESMF_StateAdd(exportState, bundle, rc)
```

OpenMI:

```
libXBeach.init();
double tstop = 0; // will be updated
libXBeach.getParameter("tstop", ref tstop);
DateTime start = new DateTime();
DateTime end = start.AddSeconds(tstop);
timeExtent.TimeHorizon = new Time(start, end);
result = libXBeach.getParameter("nx", ref nx);
result = libXBeach.getParameter("ny", ref ny);
double[] x = new double[nx+1];
double[] y = new double[ny+1];
result = libXBeach.getIddoublearray("xz", ref x);
result = libXBeach.getIddoublearray("yz", ref y);
ElementSet elementSet = new ElementSet("zb");
```

Coupling in 5 steps:

Make a library: Build a dll or shared object exposing the initialization, step, finalize. Throw out your time loop "Don't call us we'll call you!"

Allow introspection: Make all your data available through getters and setters. Extra points if you do not use global variables.

Language barriers: Use the appropriate tools. F90: use iso_c_binding others: ctypes, swig, jna.

Stick to your domain: Keep a clear separation between domains (for example numerical, physical, coupling, spatial).

Test your code: Write tests before you start coupling models.

