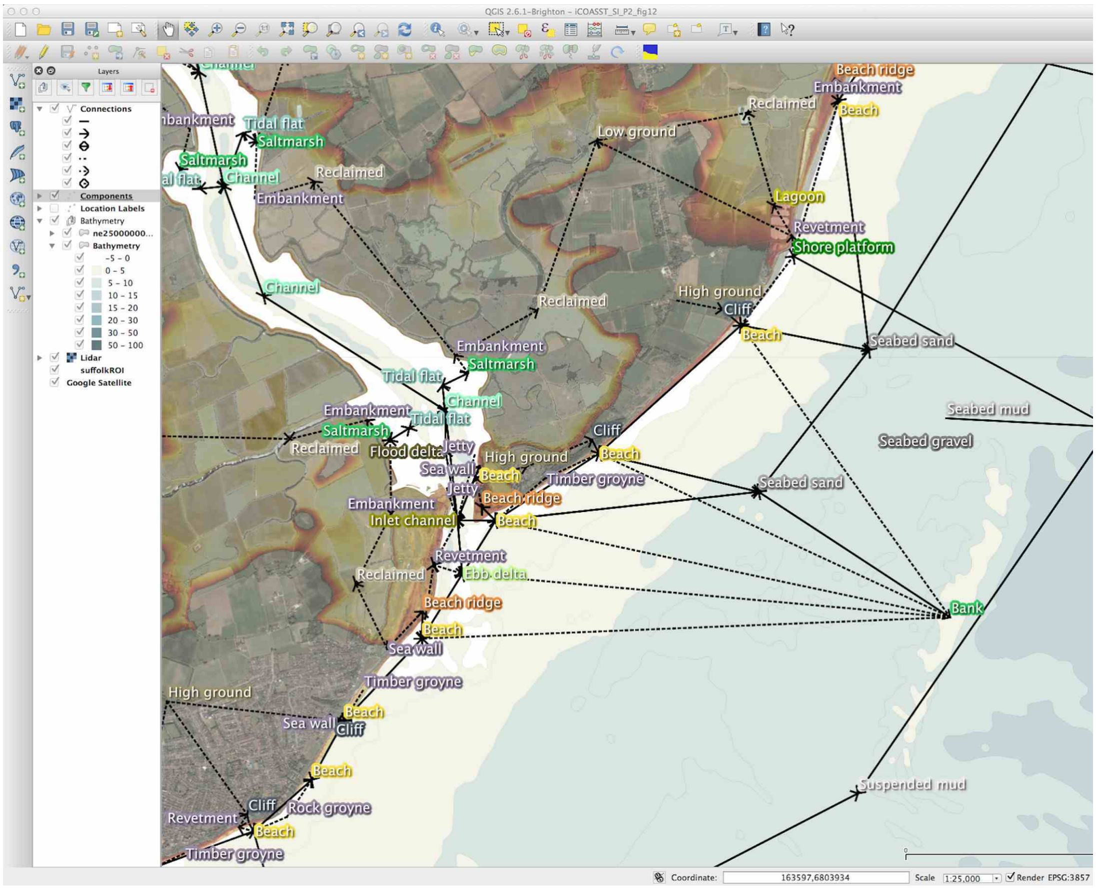
Coastal and **E**stuarine **S**ystem **Mapper**

QGIS plugin for Coastal and Estuarine System Mapping

CESMapper User Manual: release 1.5.xx



UCL Coastal and Estuarine Research Unit

CESMapper User Manual

Citation:

Thornhill, GD, French JR, Burningham H (2016) *CESMapper User Manual: release 1.5.xx*. UCL Coastal & Estuarine Research Unit, University College London, UK.

Document version:

Date:	Version/Release:		Lead author:	Checked by:
01/10/2015	v1.0.0-rev1	(initial version)	GD Thornhill	JR French
26/07/2016	r1.5.xx-rev2	(release)	JR French	H Burningham

CESMapper software repository:

https://github.com/UCL-CERU/CESMapper

Acknowledgement:

The CESMapper plugin for QGIS has been developed as part of the Integrating COAstal Sediment sySTems (iCOASST) project, funded by the UK Natural Environment Research Council (NERC project NE/J005541/1). Further information on Coastal and Estuarine System Mapping (CESM) and iCOASST can be found at:

http://www.icoasst.net

http://www.channelcoast.org/iCOASST

The CESM approach is described more fully in:

French, J.R., Burningham, H., Thornhill, G., Whitehouse, R., Nicholls, R.J. 2016. Conceptualizing and mapping coupled estuary, coast and inner shelf sediment systems. *Geomorphology* 256: 17-35. http://dx.doi.org/10.1016/j.geomorph.2015.10.006 [open access]

UCL Coastal and Estuarine Research Unit (CERU)
Environmental Modelling and Observation Group
Department of Geography
University College London
Gower Street
London WC1E6BT
UK

Contact:

j.french@ucl.ac.uk	or	h.burningham@ucl.ac.uk
+44(0)207679 0580		+44(0)207679 0577

Contents

1 CESMapper installation

1.1 System requirements

CESMapper is written in Python as a plugin for the open-source Geographical Information System software, QGIS. Provided that a compatible version of QGIS is installed, then CESMapper should function correctly. QGIS may be freely downloaded (either as binary executable application files or source code) from www.qgis.org. Both QGIS and CESMapper may be run on Linux, Mac OS-X, and Windows operating systems.

This manual describes CESMapper release 1.5.xx, which requires that QGIS version 2.8.0 (Wien) or later is installed on the computer. This release has been tested up to QGIS version 2.14.3 (Essen).

As with any GIS, the performance of both QGIS and the CESMapper plugin will be improved on computers equipped with faster processor(s) and larger amounts of memory (RAM).

1.2 Manual installation via GitHub repository

Assuming that a fully functional and compatible version of QGIS is already installed, CESMapper can be downloaded from the public GitHub repository at:

https://github.com/UCL-CERU/CESMapper

This repository contains stable release versions (suitable for most users) as well as the current development and testing versions of CESMapper (suitable only for advanced users or developers).

Installation step 1: download the current release from the repository, unzip the archive file, and copy the CESMapper directory and its contents into the .qgis2/Python/plugins directory on your computer.

On Linux systems the plugins directory is typically found at

/home/{username}/.qgis2/python/plugins (where {username} is the user account name)

On Mac OS-X systems, this directory can usually be found at

/Users/{username}/.qgis2/python/plugins

On Windows systems the plugins directory is typically found at

C:\Users\{username}\.qgis\python\plugins

CESMapper releases also contain a Docs directory (User Manual, possibly supplemented by other documentation), an Ontologies directory (see sections 2.2.3 and 3.3), and a Styles directory (QGIS style files; see section 3.9). These should all be copied to a convenient location on your computer.

Installation step 2: Start QGIS and open the Plugins … Manage and Install Plugins menu from the upper menu bar (Fig. 1.1). Locate the CESMapper plugin on the list of plugins, and check the 'install' box. Close the menu and you should see the CESMapper icon displayed on the main QGIS toolbar (Fig. 1.2).

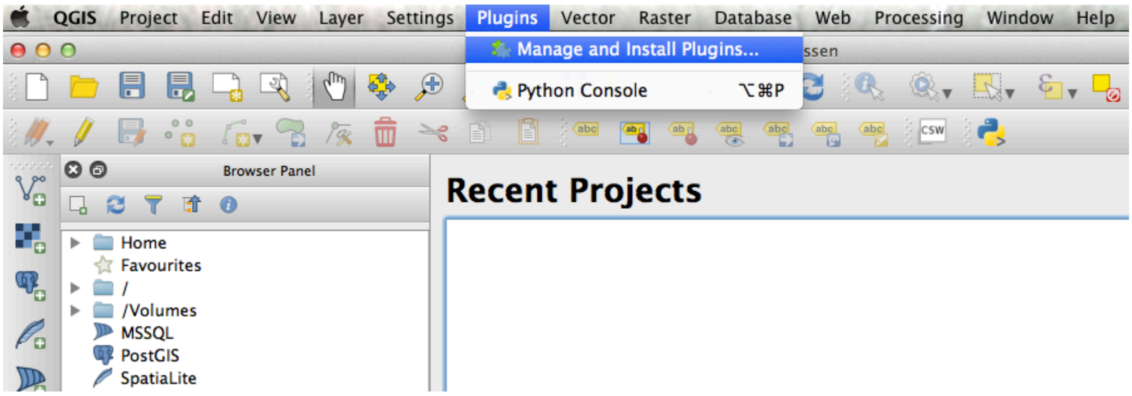


Figure 1.1 Plugin … Manage and Install Plugins on the QGIS upper menu bar.

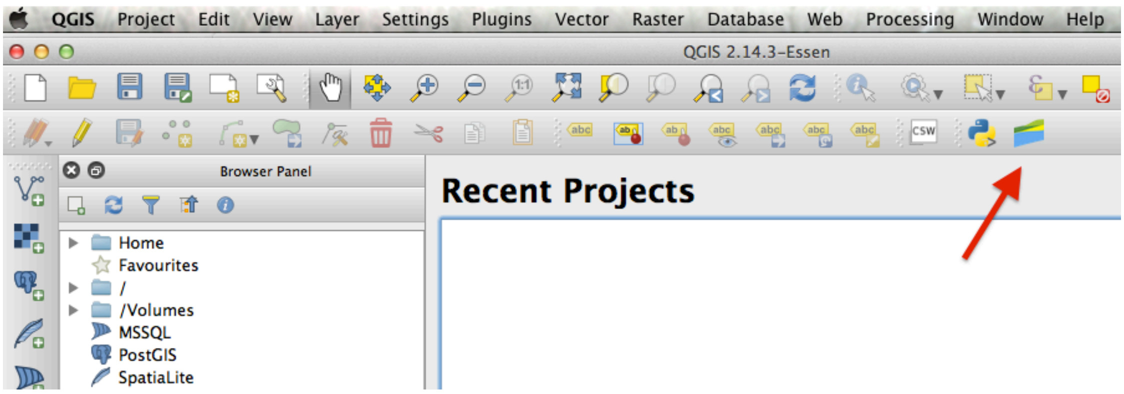


Figure 1.2 CESMapper icon (arrowed) on the QGIS toolbox indicates sucessful installation of plugin.

1.3 Installation via QGIS plugin repository

CESMapper is also being made available via the QGIS repository. This is accessible via the Plugins …. Manage and Install Plugins menu from the upper menu bar (Fig. 1.1). Locate the CESMapper plugin on the list of plugins, and check the 'install' box.

Close the menu and you should see the CESMapper icon displayed on the main QGIS toolbar (Fig. 1.2).

1.4 Software license

CESMapper is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or any later version.

CESMapper is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License (Appendix 1) for more details.

You should have received a copy of the GNU General Public License along with CESMapper. If not, see <http://www.gnu.org/licenses/>.

2 CESM overview and key concepts

2.1 Background

Risks to human populations, assets and activities are anticipated to increase with sea-level rise and climate change. Our ability to mitigate these risks is dependent on our understanding of how coastal landforms evolve their morphology in response to factors such as sea-level rise, the constraints imposed by widespread engineering structures and interventions in the sediment pathways that connect and sustain landforms. From a risk management perspective, landforms function as important mediators of the *pathways* that transfer erosion and flood risk from external *sources* (wave, tide and storm surge etc.) to human *receptors* in vulnerable areas (Sayers et al., 2002; Batten et al., 2015).

Since the pioneering work of Inman and Frautschy (1966) on the coast of southern California, the concept of the littoral cell has provided a framework for analyses of coastal change and for shoreline management. At regional to national scales, hierarchies of cells provide a geomorphological basis for shoreline management that has many advantages over the use of arbitrary administrative boundaries (Komar, 1996). In England and Wales, for example, national mapping of major cells and sub-cells (Motyka and Brampton, 1993) provided the basis for a first generation of Shoreline Management Plans (SMPs) (Cooper et al., 2002). In Western Australia, Eliot et al. (2011) have devised a three-tier hierarchy of cells to inform marine and coastal planning.

The littoral cell concept also has its limitations, however. The criteria used to delimit cells and the stability of their boundaries under variable wave climate or sediment supply are not always clear-cut. Also, littoral cells primarily reflect short-range transfers of non-cohesive ‘beach-grade’ material (i.e. sands and gravels). They are thus not well suited to handling broader scale linkages between estuarine, coastal and offshore systems, especially where longer-range suspended sediment (mud or fine sand) fluxes are important (Cooper and Pontee, 2006). Finally, the littoral cell framework has also tended to perpetuate the separate treatment of open coast and estuary in shoreline management planning (French et al., 2016b).

A behavioural systems approach to the segmentation of the open coast was devised in the FutureCoast project (Burgess et al., 2004). This embedded littoral cells within a spatial hierarchy of geomorphological units (effectively individual landforms), shoreline behaviour units (sub-systems, such as embayments and estuaries) and regional coastal behaviour systems. This approach was applied to the entire coast of England and Wales, where it has provided the basis for a second generation of SMPs.

Another advance has been the concept of the ‘coastal tract’, developed by Cowell et al. (2003a,b). This is based on the idea of a sediment-sharing system that encompasses not only the upper shoreface of the open coast but also estuarine (backbarrier) environments and the lower shoreface. The tract sits at the top of a hierarchy (or ‘cascade’) of sediment-sharing systems that evolve their morphology over progressively shorter timescales. The tract is defined at a scale at which low-order progressive change (which presents some of the most challenging erosion and flood risk management problems) can be disaggregated from higher-order variability. It also acknowledges the interactions between estuary, open coast and the inner shelf. Although the time scales of the tract hierarchy are explicit, the associated spatial scales are largely implied through the definition of morphological complexes, units and elements.

The need for an integrative systems-based perspective has become more pressing as the strategic application and evaluation of management and engineering options has evolved to address the broader time and space scales at which progressive shifts in shoreline position (Nicholls et al., 2013), and possibly overall coastal configuration, may be expected in the face of climate change and sea-level rise (French and Burningham, 2013). Of particular importance is the integration of open coast and estuary into a common conceptual framework. At the same time, there is also need for frameworks that are transparent, and which bridge the gap between the experience and perspectives of scientists, engineers and stakeholders in areas of environmental management that have often proved politically contentious. Communication is vital given the extent to which science has become almost wholly founded on models. In climate science, public understanding and confidence has often been impaired by poor communication of the nature and purpose of complex simulation models (Hall et al. 2014).

Qualitative modelling has a clear role as a means of arriving at shared understanding of the system being studied and the nature of the problems that need to be addressed (e.g. Sano et al. 2014). Coastal and Estuarine System Mapping (CESM; French et al., 2016a) has been developed within the Integrating Coastal Sediment Systems (iCOASST) project which was funded by the UK Natural Environment Research Council (NERC) between 2012 and 2016 (Nicholls et al., 2015). Within iCOASST, CESM was used for identifying the most important processes (and associated management issues) to be included in more quantitative modelling studies (van Maanen et al., 2016) and also as means of aligning stakeholder and scientific knowledge and fostering a 'participatory modelling' approach (Voinov and Bousquet, 2010; French et al., 2016c).

CESM renders the complexity of coastal and estuarine geomorphological systems as a simple ontology of components and interactions, and depicts these in a visual form that provides an effective catalyst for discussion and debate between scientist, stakeholder agencies and organizations, and local citizens. The CESMapper software is intended to facilitate application of this approach within the freely available open-source GIS, QGIS.

2.2 Key concepts

2.2.1 Idealized spatial ontology for coastal and estuarine landforms and human interventions

The starting point for CESM is an idealized spatial ontology that provides a basis for mapping the configuration of coupled assemblages of landforms on the open coast, estuaries and inner shelf. The term ontology refers to a formal specification of a conceptualization (see Gruber, 1993), but CESM adopts a fairly loose interpretation that encompasses a hierarchical classification of components and a set of permitted interactions between them. Put simply, the ontology is a classification and a set of rules.

As outlined in Figure 2.1, the CESM spatial ontology reflects some aspects of the coastal tract concept (Cowell et al., 2003a) in that it depicts a hierarchy of morphologically active sediment sharing landform systems. These are located within the geological context of a coastal shelf that can be considered time-invariant at the decadal to centennial timescales that are most relevant for shoreline management (French et al., 2016a). In contrast to the largely timescale-based hierarchy of the coastal tract, however, CESM emphasizes the spatial nesting of landform components within aggregate landform complexes (cf Burgess et al., 2004; Eliot et al., 2011). It also explicitly represents varied human interventions and the way in which these constrain landform adjustment (see, for example, Hapke et al., 2013).

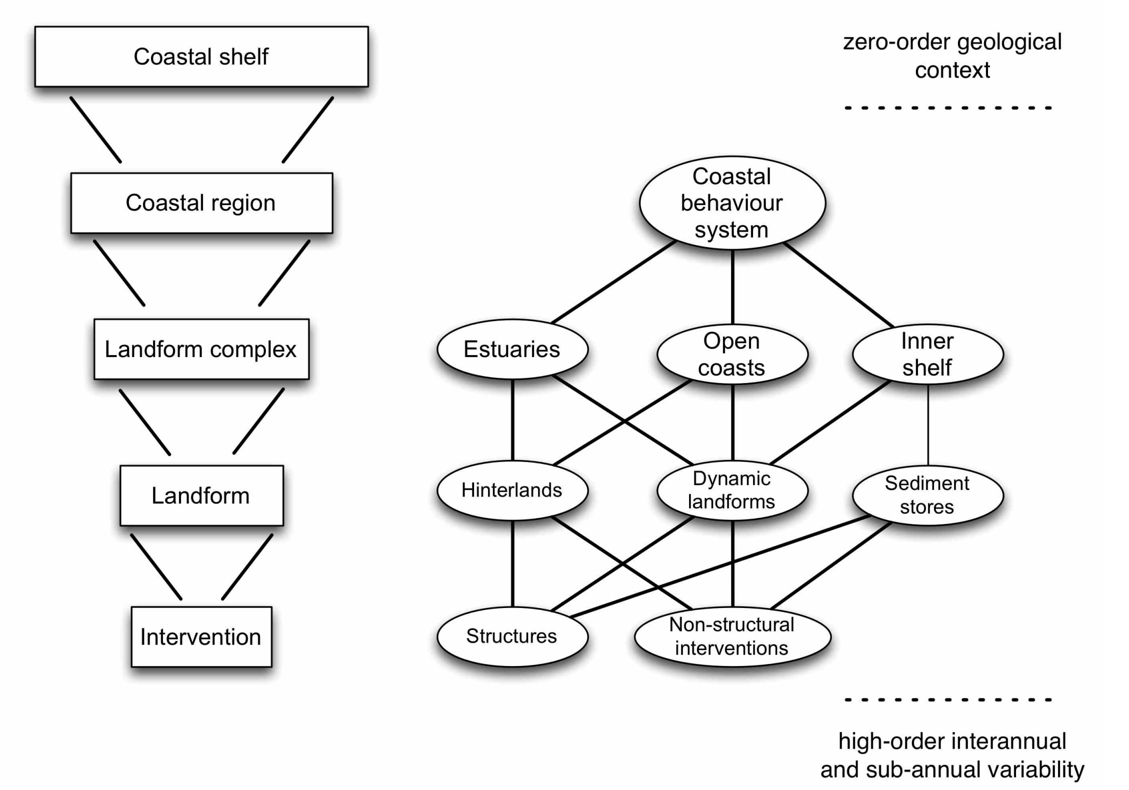


Figure 2.1: Spatial ontology of coupled estuary–coast–inner shelf geomorphic systems, showing nesting of landforms and landform complexes within broader-scale coastal regions. At decadal to centennial scales, the coastal behaviour system integrates the interaction of estuarine, open coastal and inner shelf morphodynamics, within a coastal shelf context that evolves only at much longer timescales. Interannual and sub-annual dynamics are generally not resolved at timescales of decades and longer.

Landform complexes

Estuarine, coastal and inner shelf landform complexes can be classified with reference to existing schemes and the range of landforms encountered in a given regional or shelf context. Figure 2.2a shows a simple estuary classification based on ABPmer (2008). The term ‘inlet’ is here used to define systems in which fluvial influence is negligible and sediments are purely marine in origin; this includes inlets that may be only intermittently active. Such a scheme has quite broad applicability within temperate zones (such as northwest Europe). Its relative simplicity is advantageous from a helps to reduce the ‘operator variance’ that inevitably arises where classificatory judgements have to be made.

Open coasts can similarly be classified (Fig. 2.2b) with headlands and bays for coasts that exhibit more obvious geological control. Cuspate forelands and spits may be large enough to be afforded the status of a landform complex. Barrier islands are also included as a landform complex in their own right and distinguished from other forms of non-detached coastal barrier.

The inner shelf is less replete with discrete landform types, although the drowned palaeo-landscapes of the last glacial are attracting increasing attention. Many shallow shelf seas are characterized by distinctive bank systems of varying morphology and origin. These may exert a significant influence on contemporary shoreline behaviour, either through their role in modifying wave climate (e.g. Chini et al., 2010) or via their participation in coastal sediment pathways. Figure 2.2c presents a simple classification based on work in the North Sea by Dyer and Huntley (1999).

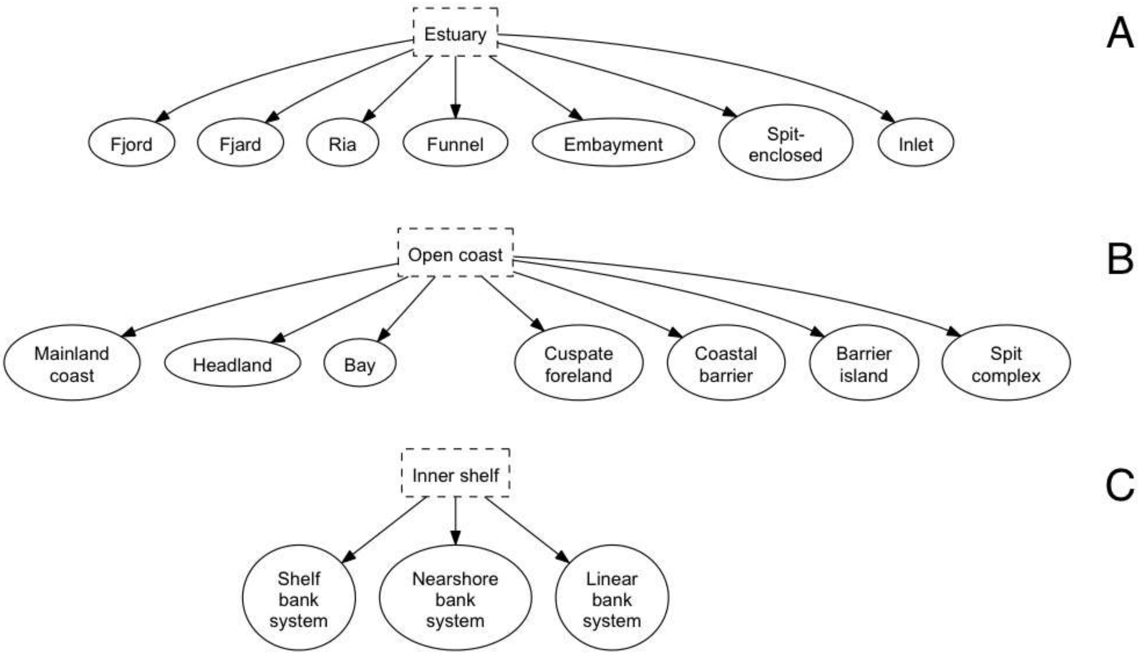


Figure 2.2: Illustrative classification of estuary, coast and inner shelf landform complexes suitable for application within a temperate (e.g. northwest European) context. The basic approach can easily be modified to suit other environments.

Landforms

The estuarine, open coastal, and inner shelf complexes outlined above represent aggregations of landforms. Table 2.1 summarises a basic set of landforms applicable to temperate settings, which includes ‘textbook’ features such as cliff, beach, tombolo, spit etc. Some landform types may occur within more than one type of landform complex (e.g. tidal flat, which can occur in both open coast and estuarine settings). Other landform types such as spits and ebb tidal deltas, occur at the interface between estuary and open coast and, as such, could be considered to be part of either complex. Spits are a special case in that larger examples can be mapped as a complex (with component dune, beach, beach ridge etc.) whilst minor features can be considered as discrete landforms within another complex. This clearly involves an element of subjective judgement.

The set of morphologically active landforms is supplemented by a smaller set of hinterland types that are considered to exert a static boundary condition control. High ground is defined subjectively as terrain that rises well above current and projected future tide and surge elevations and which would be expected to exhibit a predominantly erosional response to sea-level rise. Low ground is identified as being more susceptible to inundation, and this may constitute a more significant hazard (though erosion also leads to increased flood risk so the two hazards are not independent). Reclaimed areas are those that have been historically converted from the intertidal and subtidal zones and are protected from tidal action by defences.

Sediment systems include distinct stores of sediment that can be locally important in mediating landform behaviour. Much of the shelf is veneered by patches of sediment, some of which are essentially inactive under current sea level, wave climate and tide regime, and some of which participate in sediment pathways that interact with coastal or estuarine environments.

Table 2.1: A basic set of landform components common to open coastal, estuarine and inner-shelf complexes. These comprise morphologically active landforms, as well as hinterlands and sediment stores.

Landform		Hinterland	Sediment store
Cliff	Inlet channel	High ground	Seabed gravel
Shore platform	Ebb delta	Low ground	Seabed sand
Beach	Flood delta	Reclaimed	Seabed mud
Beach ridge	Bank		Suspended mud
Tombolo	Channel		
Dune	Tidal flat		
Spit	Saltmarsh		
Rock outcrop	Brackish marsh		
Lagoon	River		

Human interventions
Present-day coastal behaviour is strongly conditioned by, and indeed partly a consequence of, human interventions of various forms over a period of decades to centuries. The effects of coastal protection works are evident locally (Basco, 2006), regionally (Brown et al., 2011) and even nationally at the scale of the USA (Hapke et

al., 2013). The most obvious interventions have been structural, with the goal of erosion control, reclamation or flood risk reduction. Local experiences and requirements, and has given rise to a diverse terminology for interventions that perform the same basic function. Table 2.2 presents a highly generic classification of some basic types of intervention according to the function performed. Most have the effect of arresting shoreline movement, for example by limiting erosion or channel migration. Non-structural interventions in coastal and estuarine sediment systems are also pervasive, not only through dredging and aggregate extraction but also through the adoption of ‘softer’ approaches to coastal management.

Table 1.2: Minimal classification of generic structural and non-structural interventions in estuary, coast and inner shelf sediment systems, with their indicative purpose.

Structural	(indicative purpose)	Non-structural	(indicative purpose)
Seawall	Erosion protection	Dredging	Navigation; mining
Revetment	Erosion protection	Dredge disposal	Spoil disposal
Bulkhead	Erosion protection	Sediment recharge	Restoration of sediment deficit (beach, intertidal)
Embankment	Flood protection	Sediment bypassing	Continuity of sediment pathway; navigation
Barrage	Flood protection	Sediment recycling	Resilience (beach profiling);
Breakwater	Wave energy reduction		
Detached breakwater(s)	Wave energy reduction		
Groyne(s)	Sediment retention		
Training wall	Channel stabilisation / navigation		
Jetty	Varied		
Outfall	Drainage / dispersal		
Quay	Navigation/trade		
Dock	Navigation/trade		
Weir	Regulation of river gradient and/or tidal limit		

Interactions

From a functional perspective, system components influence each other through complex web of interactions. Interactions include any cause-effect relation between components; for example, a jetty exerts an effect on an inlet channel, stabilising its location and influencing its cross-sectional characteristics. A sub-set of the interaction network involves transfers of mass and these sediment pathways, taken together, define the sediment budget (Bowen and Inman, 1966; Rosati, 2005).

Some of the linkages may be simple unidirectional ones, for example where sequential beach units define a littoral drift system. Other interactions are clearly bidirectional, such as the interplay between a seawall and a beach (Basco, 2006). Others may represent more complex causality: a cliff may source sediment to a fronting beach (mass transfer) and the beach may influence the cliff (via an influence through which beach morphology feeds back into the cliff recession rate; Walkden and Hall, 2011). Table 2.3 illustrates some of these interaction types, and the distinction between pure influences and sedimen transfer pathways.

Table 2.3: Illustrative paired examples of system interaction rules for landforms and interventions.

From	To	Interaction	Logic (literature source)
Cliff	Beach	Sediment pathway (sand, gravel)	Cliff sources beach-grade sediment (mud typically lost offshore)
Beach	Cliff	Influence	Presence and morphology of beach feeds back into cliff recession rate (e.g. Walkden and Hall, 2011)
……	……	……	……
Seawall	Beach	Influence	Presence of seawall may cause lowering of beach (e.g. Basco, 2006)
Beach	Seawall	Influence	Beach protects toe of seawall and reduces wave energy on face
……	……	……	……
Jetty	Inlet channel	Influence	Jetty exerts stabilising influence on channel position and constrains width adjustment
Inlet channel	Jetty	none	No direct causal relation in this direction

2.2.2 Mapping the connectivity of coupled coast and estuary systems

CESMapper builds on earlier proof-of-concept work (French and Burningham, 2009; Whitehouse et al., 2009) to define a workflow for mapping the connectivity of coupled coast and estuary systems using the ontology described in the preceding section. This workflow is summarised in Figure 2.3 and involves three main stages:

1. **Specification** of the problem or application
2. **Mapping**, in which the system is conceptualized as a spatial network of interactions
3. **Augmentation**, in which data or meta-data are appended to the system map.

The current release of CESMapper is mainly concerned with the Mapping stage, although future versions will likely support the creation of spatial databases that build on the system map.

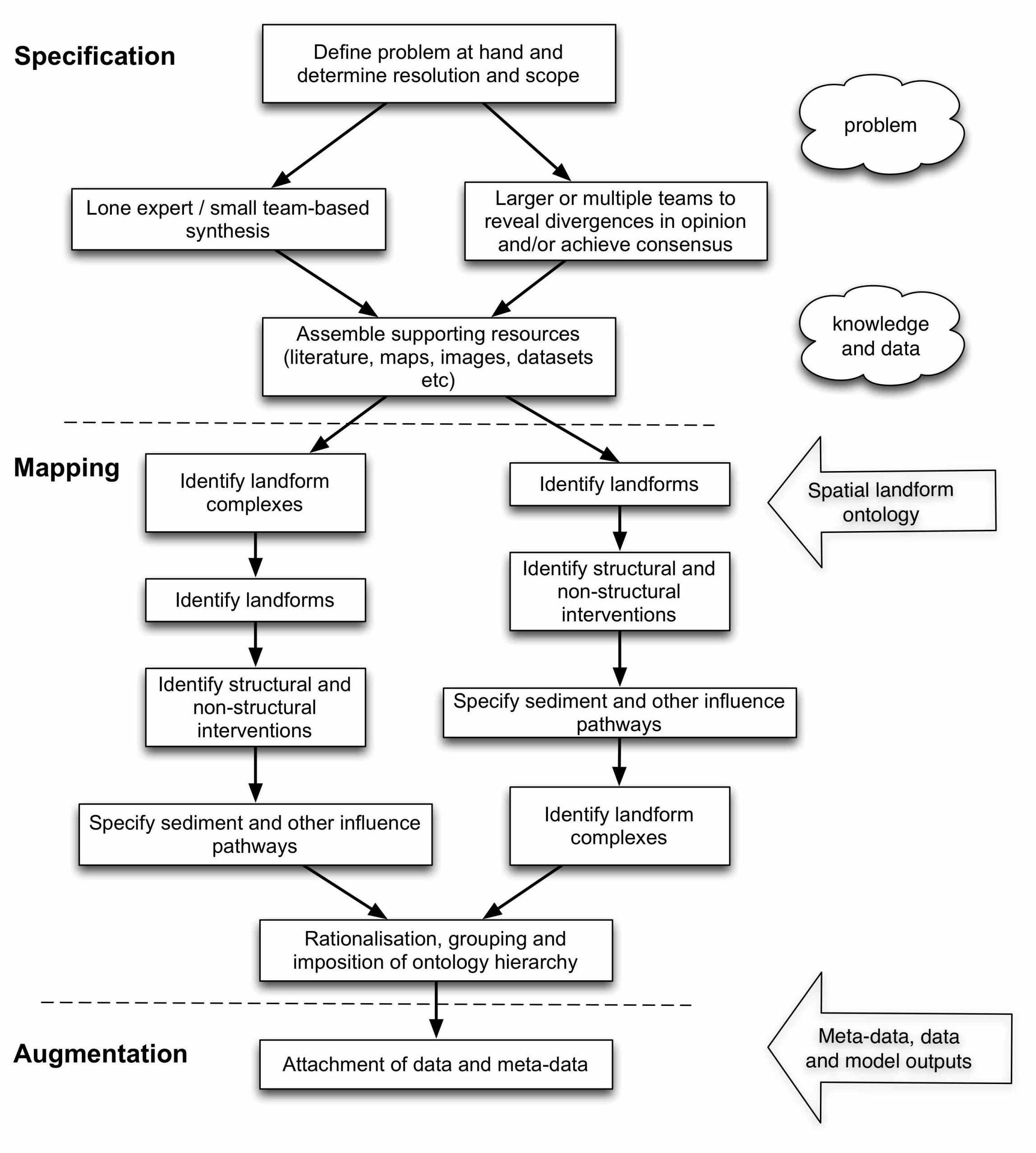


Figure 2.3 – Workflow for Coastal and Estuarine System Mapping. CESMapper is mainly concerned with the Mapping stage.

Key stages of the workflow include:

1. Specification of the problem. This starts with a judgement of the appropriate time-averaging period over which to characterise the system, an appropriate spatial resolution, and the geographical scope. The scope might vary from regional mapping to guide preparation of a shoreline management plan, to a more detailed local map as context for an engineering scheme.

For well-documented and/or understood systems, a lone ‘expert’ or small team of ‘experts’ may be able to achieve a relatively uncontentious synthesis of existing knowledge. Where the system is less well understood, CESM provides a starting point for the progression of a conceptual model and a larger team might be required to achieve a consensus. In either case, participatory involvement of stakeholders may be beneficial.

Once the problem is defined, background knowledge (published papers, reports etc.) and datasets (aerial images, geological maps, bathymetry etc.) are drawn together to inform the mapping process.

2. System mapping. Mapping is undertaken using the hierarchical set of landforms and interventions defined in the ontology. Customization of the ontology will usually be required to suit particular geographical contexts and applications and a feature of CESMapper (see section 2.2.3) is that this can be done via external library files without any modification of the software.

Landforms and interventions are first mapped in detail and then organised into broader-scale complexes. Mapping of the open coast proceeds by identifying distinct hinterland – backshore – nearshore sequences and any local constraints due to structures or known non-structural interventions (e.g. beach nourishment or sediment bypassing programmes). This is similar to the approach taken by Hanson et al. (2010), who presented a scheme for mapping barrier and non-barrier coasts based on sequential transitions in cross-shore profile type, as defined by a set of prescribed landform elements. Within an estuary, distinct subtidal – intertidal – hinterland transitions are similarly mapped with reference to the dominant axis of the estuary.

Interaction pathways are then added, with the directionality of the sediment pathways indicated, and distinction made between these and ‘influence only’ interactions (e.g. those involving the various structures) that are not part of the sediment system. Sediment pathways will often have a preferred direction, but may also be bi-directional where movements are uncertain or oscillatory.

An illustrative conceptualisation of open coast – estuary interaction over an oblique Google earth view is shown in Fig. 2.4.

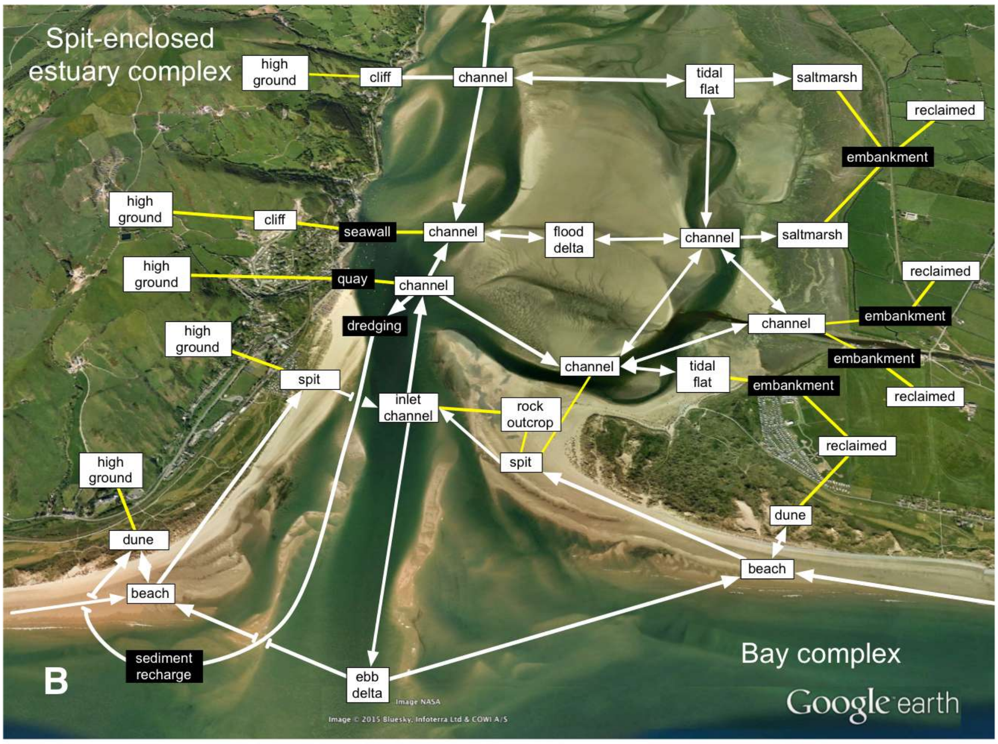


Figure 2.4 Illustrative conceptualisation of open coast – estuary interactions at Aberdovey (Wales, UK). Base image courtesy of Google Earth. Note that this mapping is purely illustrative and not intended to be interpreted as a detailed case study.

3. Augmentation. System maps can be annotated to include metadata (e.g. references and links to relevant research and datasets) as well as data (e.g. quantitatiive sediment flux estimates). CESMapper exports the system map as standard GIS shapefiles and the attributes of these can be modified within QGIS. Future releases of CESMapper will provide a more direct means of appending data.

2.2.3 CESMapper software

The CESM approach has its origins in proof-of-concept work undertaken as part of an Environment Agency-funded project on large-scale coastal geomorphological behaviour (French and Burningham, 2009; Whitehouse et al., 2009). Mapping was initially undertaken using concept mapping software (CmapTools; Cañas et al., 2005). CESMapper provides similar functionality within the geospatial framework provided by a Geographical Information System (GIS).

QGIS (http://www.qgis.org) was selected as a preferred geospatial platform on account of its maturity as an open source GIS, support for multiple operating systems and growing user base. QGIS is written in C++ and allows integration of software plugins coded in either C++ or Python. CESMapper is implemented as a Python plugin that enables system components to be mapped interactively over one or more QGIS data layers.

CESMapper allows the user to add landforms, interventions and interactions on top of a GIS base layer that defines the projection and co-ordinate system. Possible base layers include digital mapping, Web Map Server-based layers (including Google Maps or Bing maps), or digital photography. The base layer can be supplemented by 'helper layers' that provide useful information to guide the identification of landform types and identify human interventions. Airborne LiDAR raster layers are especially useful, as are digital bathymetric charts and geological maps, and vector layers containing information on flood and coastal defence infrastructure.

A key feature of the CESMapper architecture is that the landform ontology is completely separate from the software itself and resides in a user-editable external library file. The ontology is defined using a simple XML-like semantic markup language, which permits the inclusion of optional presentational markup to impose various label and line style settings (these can be overridden within the software). The available components (landforms, interventions) are read from the ontology and used to guide on-the-fly creation of Graphical User Interface (GUI) palettes, which provide the user with a pre-determined set of system elements and impose constraints on how these can be combined. CESMapper then allows the linkages between the various components and specify the type and directionality of the connection (influence, sediment transfer) to be defined. CESM is provided with an ontology devised during the iCOASST project (contained in the file iCOASST-CESMOv1.7.txt) that is intended mainly for application in the UK. Applications outside the UK may require this file to be customized by the user (see section 3.3 and Appendix B).

The selection of a combination of landforms to be included as part of a specific landform complex can be accomplished using the software tool which will automatically provide a check that the grouping is permissible within the defined ontology; this maintains a base level of consistency between different users when producing coastal and estuary system maps. The resulting map (a point layer of components and a line layer of connections) is saved in ESRI shape file format, which can be read by a wide variety of other applications and thus provides a common platform for distribution of system maps to stakeholders.

2.3 Applications of CESM and CESMapper

Within the iCOASST project (Nicholls et al., 2015; Van Maanen et al., 2015), coastal and estuarine system maps were well received by a diverse group of stakeholders that included, inter alia, management agencies and regional authorities, non-governmental organisations, representatives of industry and agriculture, and local inhabitants. Discussions typically centred on matters of detail, such as the omission of local geological controls on shoreline position, as well as broader scale divergences in opinion – notably concerning the consistency of the direction of littoral drift. As a result we revaluated and reformulated our quantitative understanding on the Suffolk coasts. Hence, these discussions have been valuable in capturing stakeholder knowledge and feeding this into both data-driven analyses and modelling studies. The capturing of local knowledge and its incorporation into the formulation of a problem and an approach to it, are key elements of good modelling practice that have all too often be neglected in coastal analysis.

Importantly, CESM is transparent and accessible to a wide range of users. This is partly a consequence of its implementation in open-source software. This counters one of the major shortcomings of a 'top down' approach to shoreline management planning that has historically been heavily reliant on proprietary closed-source model codes and GIS software that is available to the larger consultancies but not to local communities and smaller consultants. As French et al (2016b) note, the open source paradigm of computer science is a good exemplar here, in that it demonstrates the benefits of a genuine community effort, both in terms of transparency and accessibility and also in terms of legacy. CESM has the potential to create conceptual models that are living community efforts, stimulating a greater sense of shared endeavour between modellers and stakeholders than has thus far been possible. These conceptual models and linked databases are free to evolve beyond individual project timelines through the continuing involvement of a community of researchers and stakeholders. Hence, the system maps constitute information products that are not finalised at a project end date but, instead, remain free to evolve as knowledge accumulates and agendas change over time.

3. Using CESMapper within QGIS

3.1 Software workflow

CESMapper implements the various stages of the CESM process set out in French et al (2016a) in a simple workflow. This is summarised in Fig. 3.1.

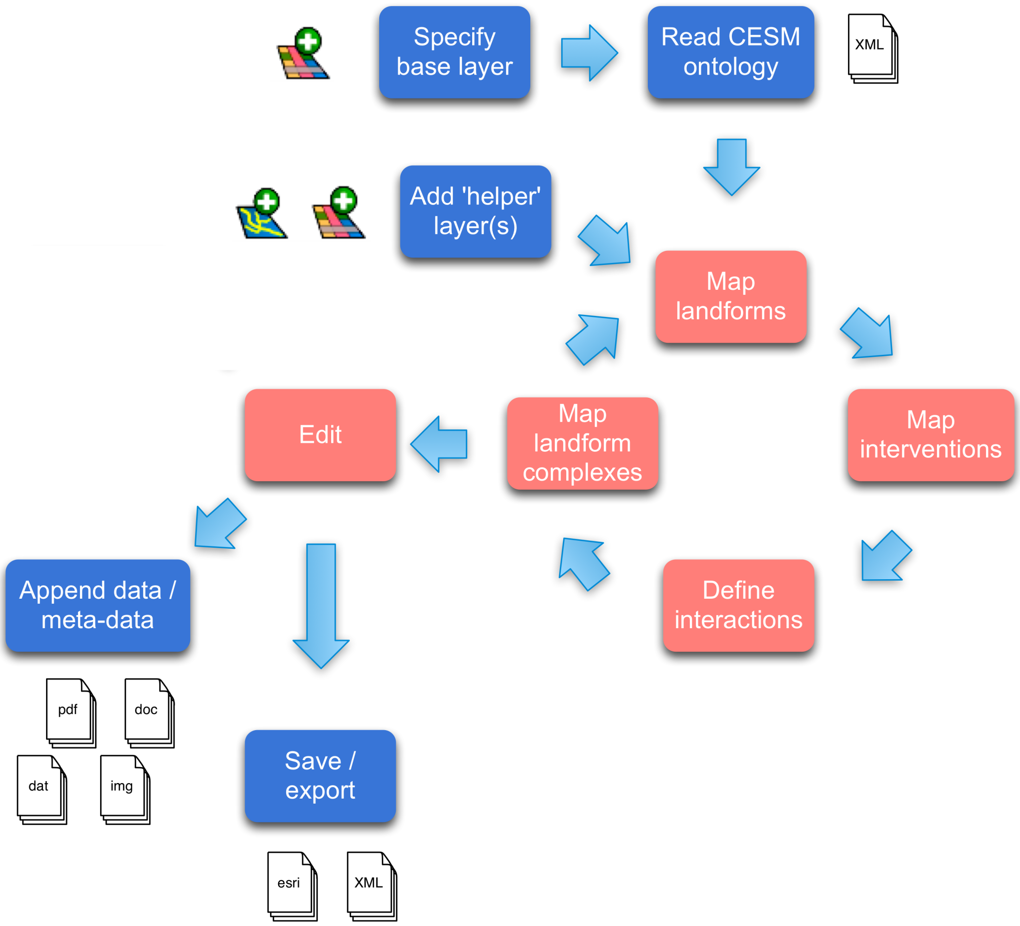


Figure 3.1 Workflow of CESMapper. At least one GIS base layer is required on which to interactively locate landforms, interventions and interactions. Helper layers (bathymetry, geology, flood defence lines etc.) can be used to inform the interpretation of the coastal and estuarine systems.

3.2 Loading a GIS base layer

CESMapper requires that at least one base layer is loaded within QGIS. If no suitable layer is present the user is prompted to load one (Fig 3.2). The base layer is important as it sets the base coordinate system and all subsequent maps will be referred to this.

It can be any raster or vector layer but is typically a chart or georeferenced aerial image. Google or Bing Maps coverages are available via the QGIS OpenLayers Plugin and are particularly useful as a basis for coastal mapping over extended areas.

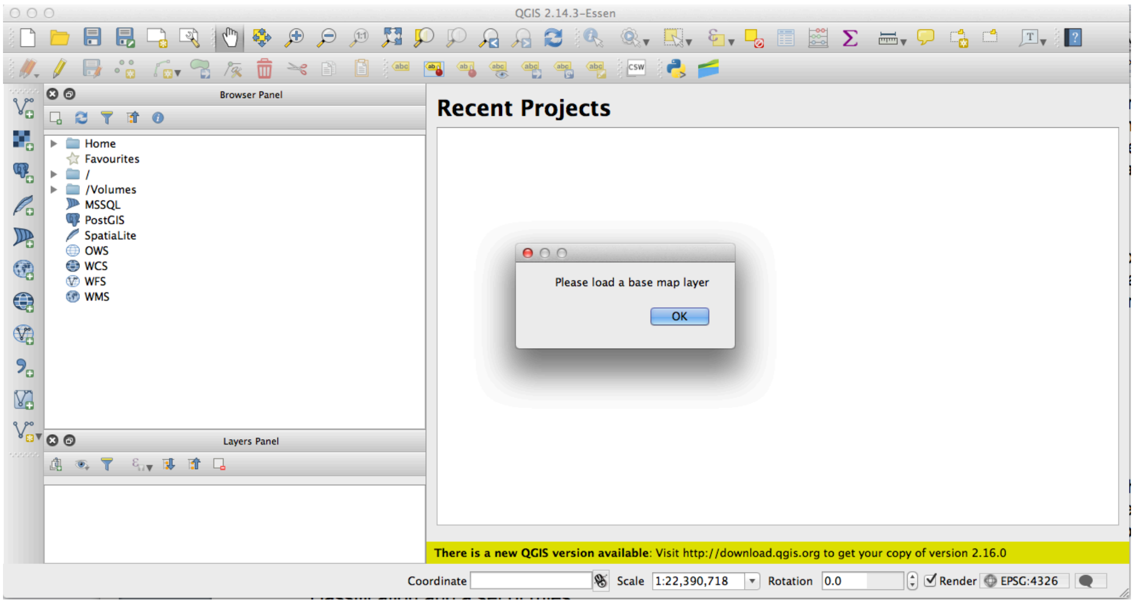


Figure 3.2 On startup, CESMapper will request that a suitable base layer is loaded into QGIS before mapping can be initiated.

3.3 Loading a CESM ontology

A key feature of CESMapper is the separation of the overarching landform ontology from the software itself. The ontology, which specifies the hierarchical set of available system components and the permitted interactions between them, resides in an external library file. This follows a simple XML-based syntax (see Appendix B). CESMapper parses the library file to generate the various interactive map creation menus. Any changes to the ontology will be reflected in the components available within the CESMapper. Because the ontology can easily be modified using a standard text editor, the user can customise the set of system components available within CESMapper.

CESMapper will automatically request that a landform ontology library be loaded (Fig 3.3). An example ontology developed for application in England and Wales as part of the iCOASST project is included with the plugin distribution. This file, “iCOASSTLibraryFilev1.7.txt”, should be copied to a suitable location on your computer.

Select and load the ontology. On completion, the main CESMapper menu will appear (Fig 3.4). This may be dragged to anywhere on your screen, and subsequent windows will open below it.

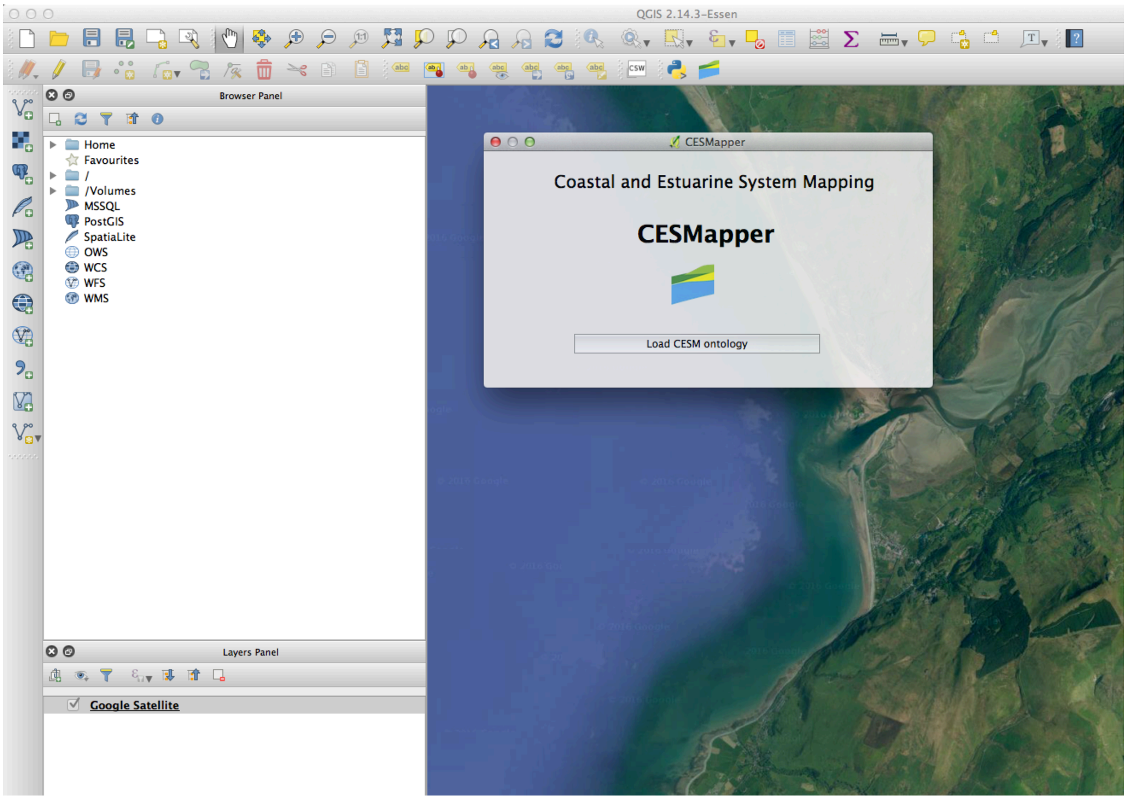


Figure 3.3 Once a base layer is loaded, CESMapper prompts the user to load a CESM ontology library.

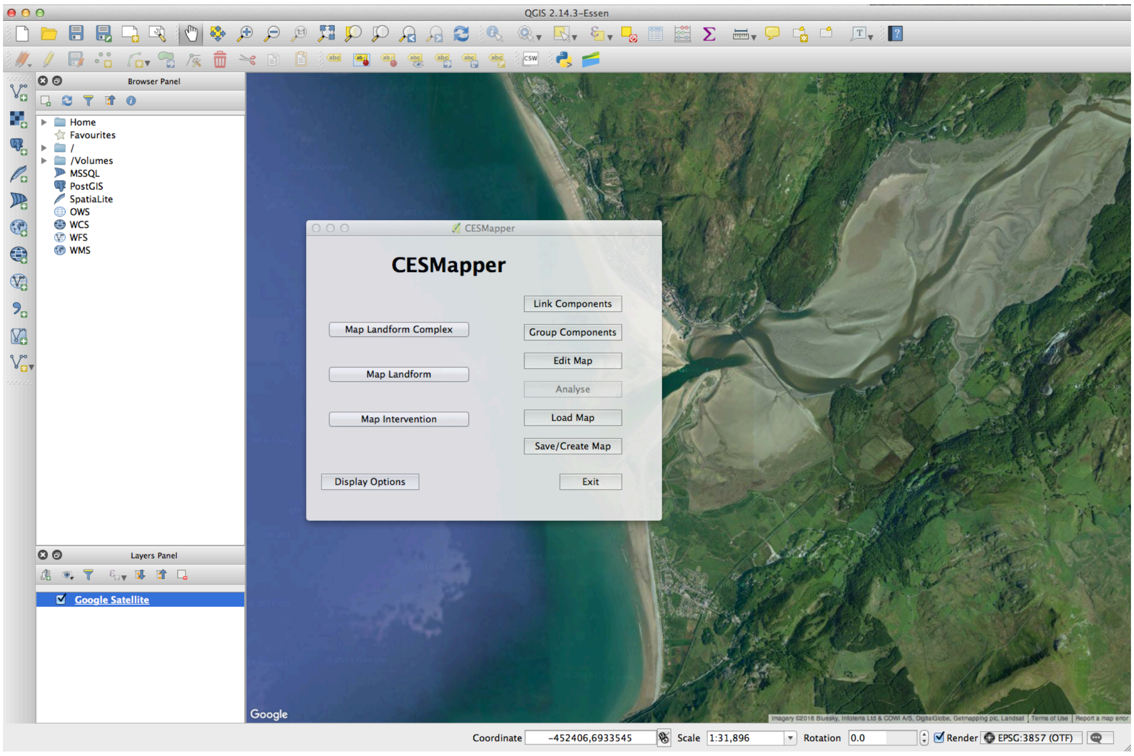


Figure 3.4 Screenshot of the main CESMapper window and tool options.

3.4 Create and save system maps

Within QGIS, the coastal and estuarine maps are created as two shape files. These follow a standard ESRI-shapefile format. One shapefile holds the set of discrete system components (landforms, sediment stores, human interventions), and the other stores the connections between them (influences, sediment pathways). Before mapping can commence, this pair of GIS shapefiles must either be created or a pre-existing map (i.e. previously saved CESM shapefiles) must be loaded.

CESMapper uses a set naming convention for these paired files. This naming convention must be observed to enable the tool to recognise and load the correct paired shape files. Assuming that *{Project}* is the project (or system map) name, the file set should be:

Component layer	Connection layer
{Project}.cpg	*{Project}*Con.cpg
{Project}.dbf	*{Project}*Con.dbf
{Project}.prj	*{Project}*Con.prj
{Project}.qpj	*{Project}*Con.qpj
{Project}.shp	*{Project}*Con.shp
{Project}.shx	*{Project}*Con.shx

To create a new map, click the “Save/Create Map” button (Fig 3.4) and in the dialogue that opens (Fig 3.5) specify a CESM map name and save to the desired location. This will create the paired shape files for both the system components and their interactions and load them as layers into QGIS. A further QGIS dialogue will appear requesting that the user specifies the Coordinate Reference System (CRS) to be used for the CESM components and connections layers.

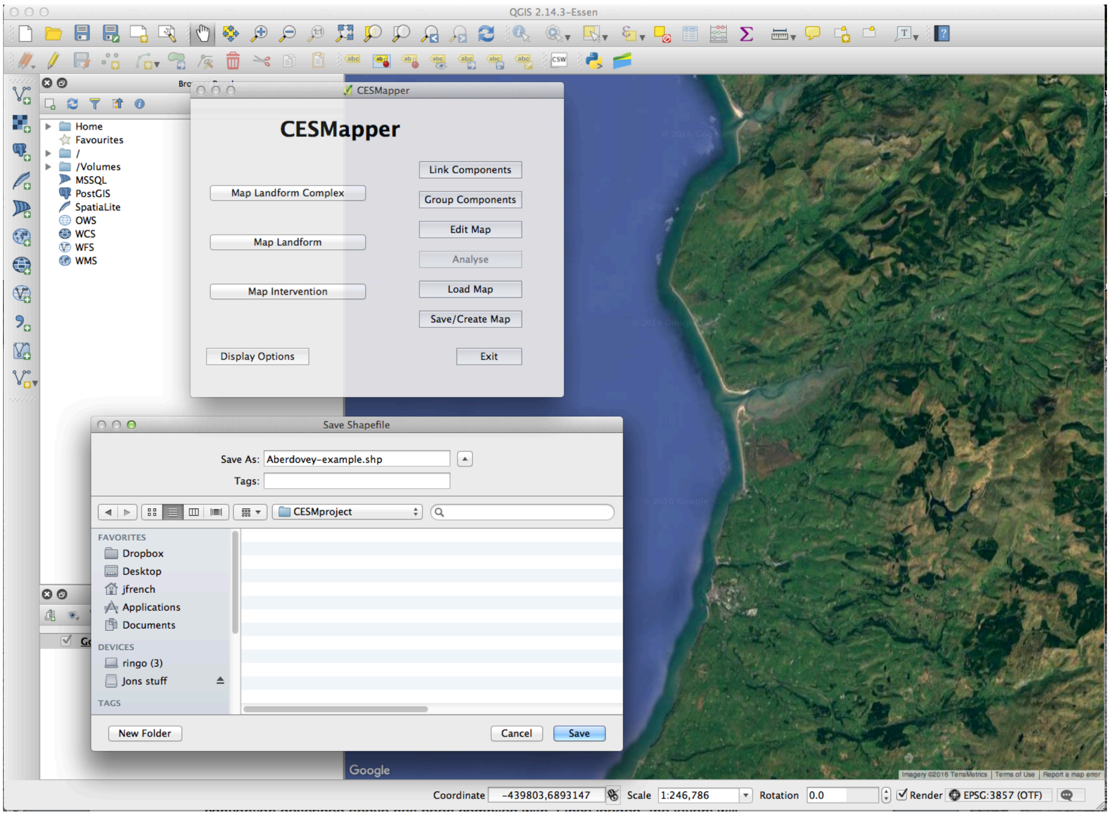


Figure 3.5 Dialogue for creation of new CESM map.

If you have previously created CESM shapefiles and wish to view or continue working on these using CESMapper, click the “Load Map” button (Fig 3.4), navigate to the main project file, *Project*.shp (where *Project* is the title of the project) and select it. The tool will automatically load the corresponding connections file, provided the naming convention described above has been complied with. Once loaded, the layers will appear in QGIS.

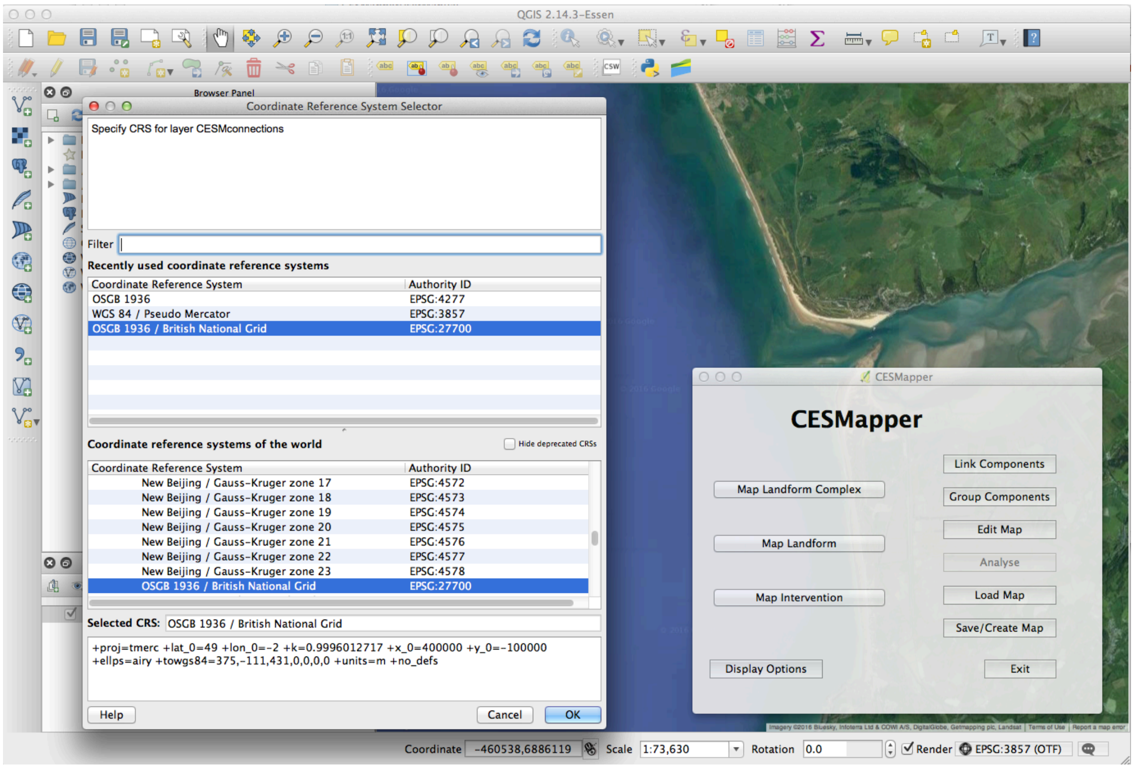


Figure 3.6 Dialogue for specification of Coordinate Reference System.

3.5 Mapping of landforms and human interventions

On the left hand side of the main CESMapper menu (Fig 3.4) are a set of buttons that allow mapping (interactive placement over the base layer) of landform complexes, individual landforms, and human interventions on the base layer. The number and naming of these hierarchical levels is defined by the external CESM ontology and so will vary if this has been customized by the user.

When one of the 'Map …' buttons is clicked, a new menu pane opens with the list of landform complexes, landforms or interventions as appropriate (Fig 3.7). Again, the exact set of components listed will depend on the CESM ontology that is loaded. Choose the component you wish to place via its radio button, then click on the base map to place it. Select and place other components as required (see example in Fig 3.8).

When you have completed mapping for this set of components, click the "Finished Mapping" button. This will save the changes to the layer, close the mapping menu, and return control to the CESMapper main menu. To ensure the changes are saved to the CESM project (i.e. the two component and connections shape files) you must click the "Save/Create Map" button. This saves the updated files.

The above process may be repeated as many times as desired. This allows a large or complex map to be built up in stages.

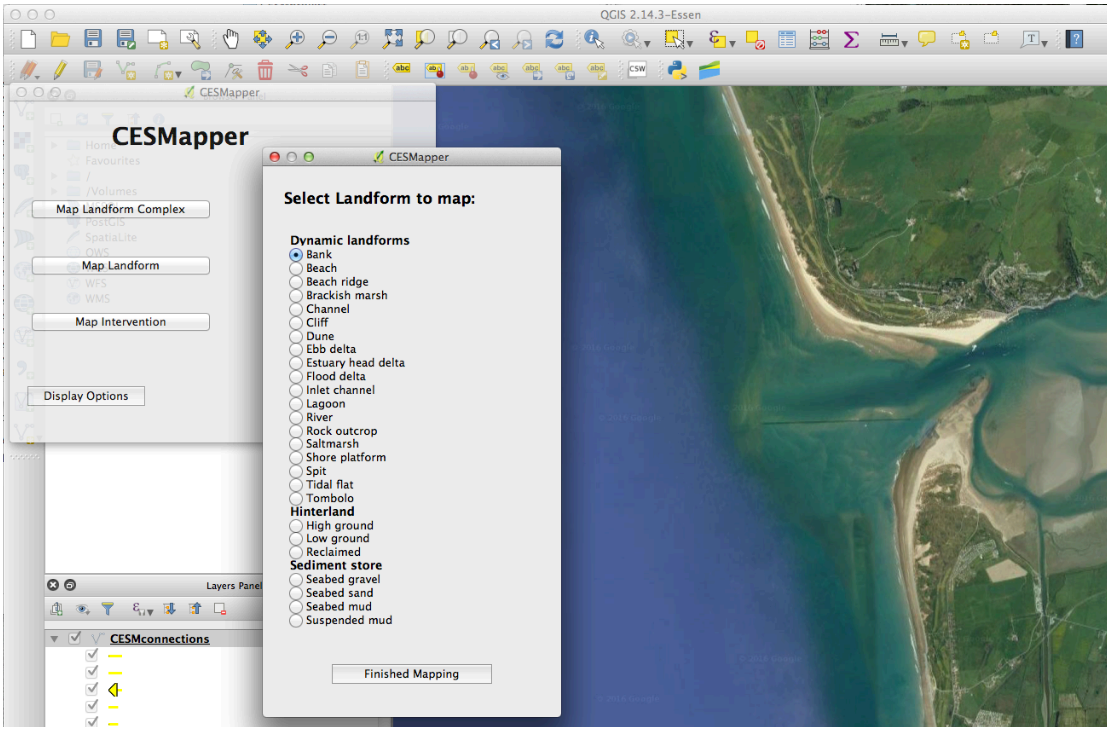


Figure 3.7 Typical menu for interactive mapping of landform components. Exact set of components available will vary according to the contents of the external CESM ontology that is loaded into CESMapper.

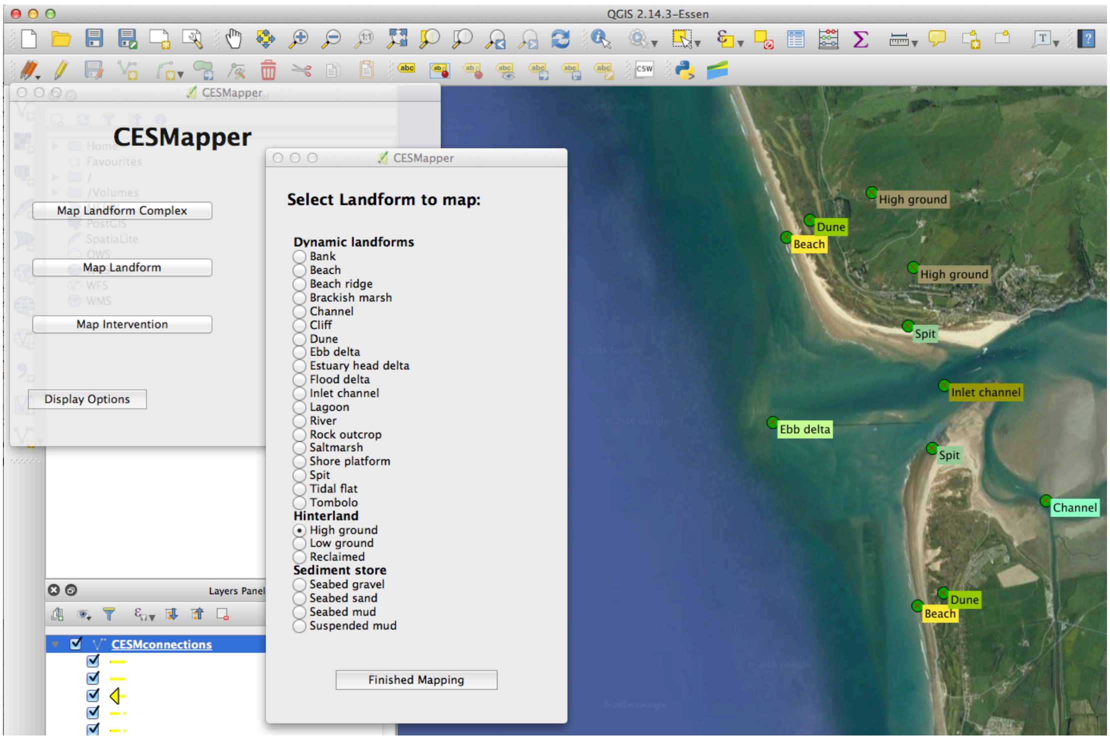


Figure 3.8 Illustration of landform mapping in progress.

3.6 Connecting system components

Once the system components have been mapped, the linking tool can be activated from the “Link Components” button. This opens a menu (Fig 3.9) that allows the user to select the type of connection (mass transfer, influence, or mixed), as well as the direction of the interaction. The direction is established by the order in which components are selected. The user can select two components (and only two components) to connect together. Any mistake at this stage can be corrected by using the “Clear Selection” button to reset the connection process. After the two components have been selected, click “Connection Completed” to add and display the connection on the system map. The type and directionality can be changed for each connection.

When you are satisfied with the connections (see example in Fig 3.10), close the component connection menu. It is advisable to save the system maps frequently whilst performing the connections (use the “Save/Create Map” button on the main CSEMapper menu, which should remain visible). This ensures that any changes are written to the shape files and avoids the loss of this information in the event of any problem with the computer or software.

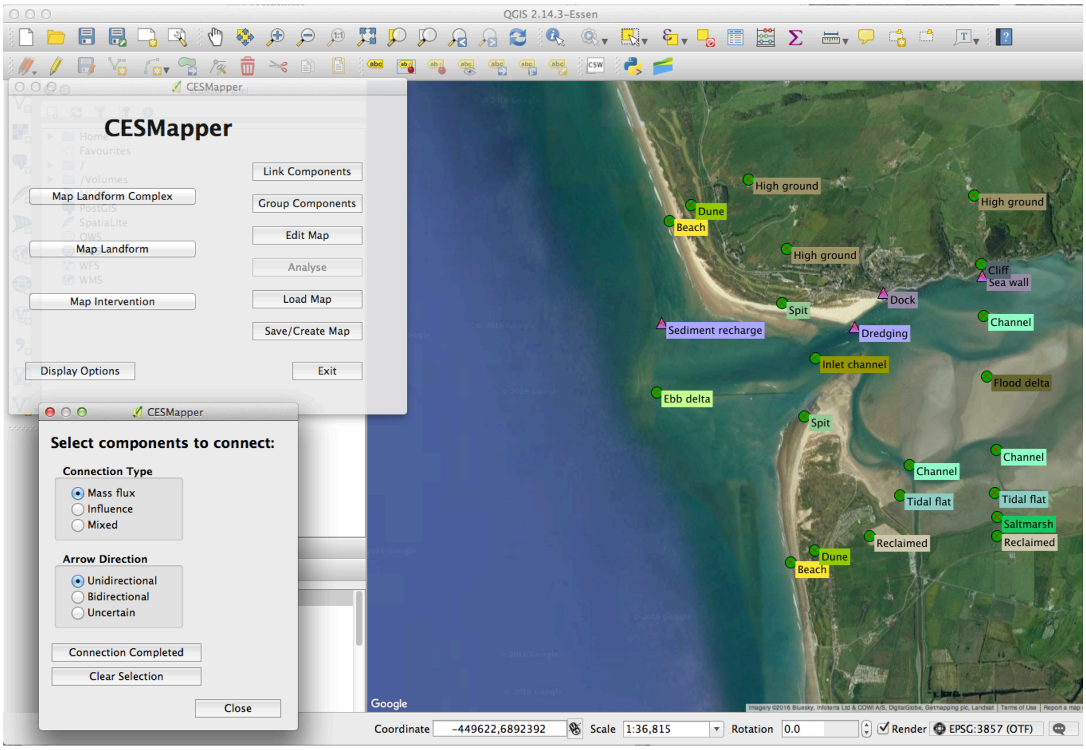


Figure 3.9 Screenshot of the component connection menu.

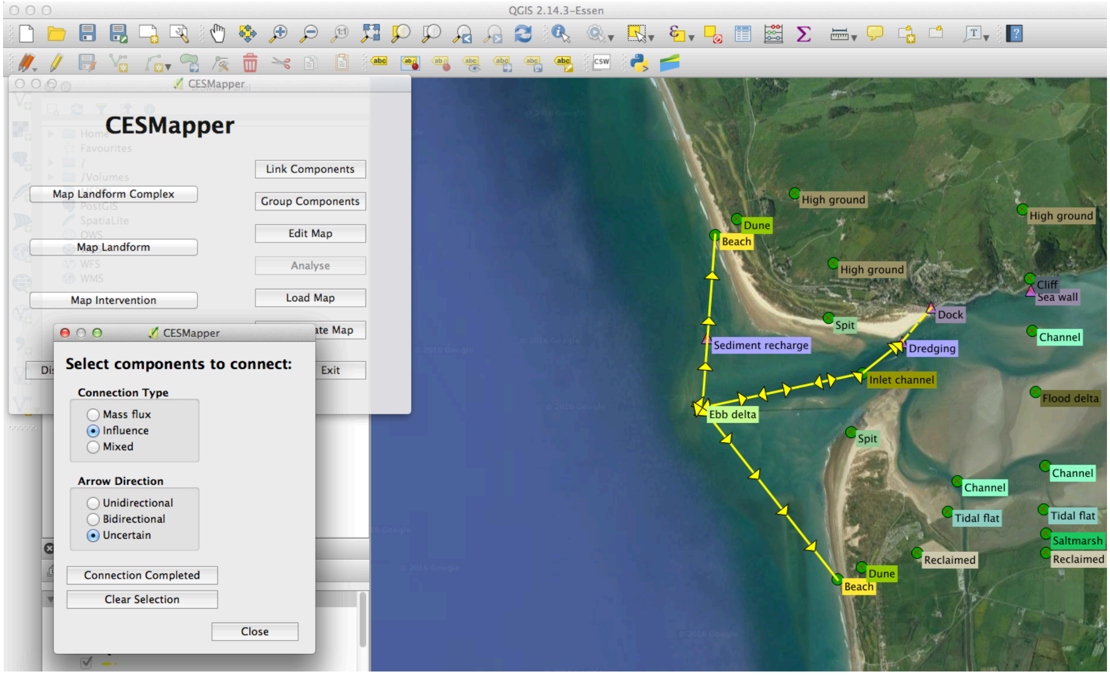


Figure 3.10 Illustration of the component connection stage in progress.

3.7 Grouping of landforms into complexes

This functionality will be completed soon in a new CESMapper release.

3.8 Map editing

If you wish to move or remove a component, it is best to use the built-in editing tool within CESMapper to ensure that the attribute tables are updated, and the links between the two component and connections layers are synchronised correctly. The “Edit Map” button opens an editing menu, which allows the user to select “Move Component” or “Remove Component” (Fig 3.11).

Select the desired action you require, then click on the component. To move a component, click and hold down the mouse button while dragging the component; any connections to that component should be redrawn to maintain the connection. If you select “remove component”, the component will disappear, along with any connections previously made to it. It is necessary to click move or remove for each such edit to ensure the edits are correctly carried out. When editing is completed, click “Done editing”, and close the menu to return control to the main CESMapper menu.

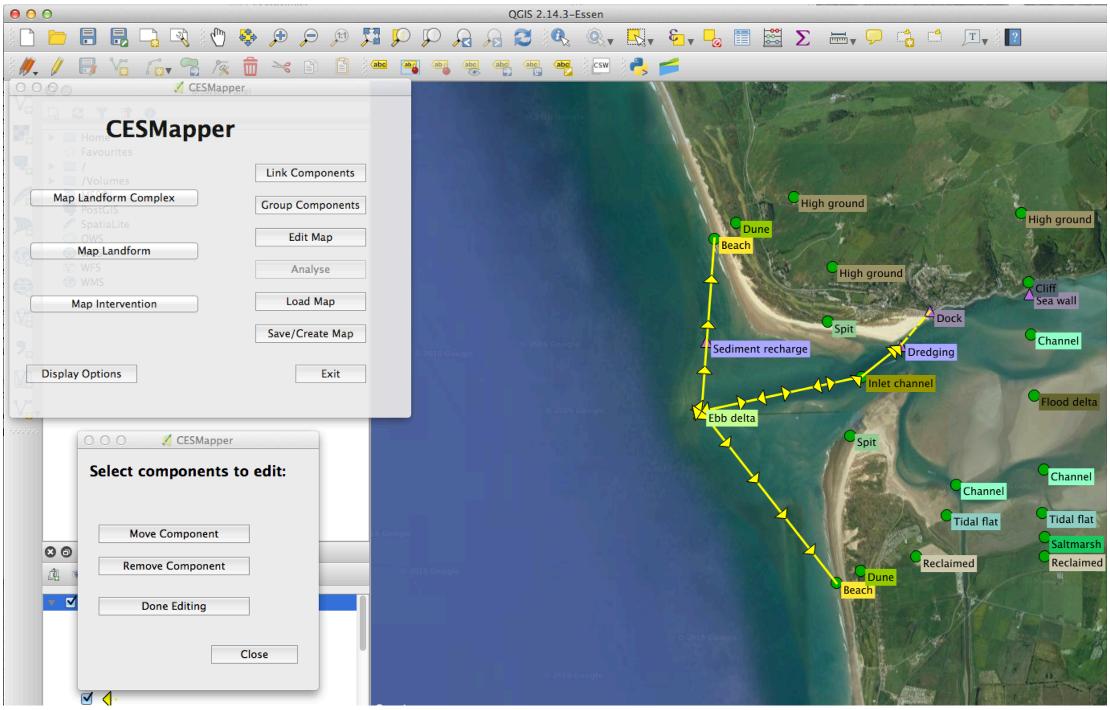


Figure 3.11 Illustration of the component editing menu.

3.9 How to load additional supporting layers into QGIS

It is often useful to load additional layers into QGIS in order to aid the interpretation of the coastal and estuarine environments being mapped. Raster layers (such as georeferenced aerial photographs, scanned bathymetry charts, or airborne lidar mosaics) can be loaded using the 'Add Raster Layer' button on the QGIS toolbar (Fig 3.12). Vector datasets can be added using the 'Add Vector Layer' button.

The CESMapper tool is distributed with style files that identify key attributes of the components and connections shapefiles to display the features clearly. Landform complexes, landforms and interventions in the components shapefile are displayed using large symbols and labelled using tags the colour of which are defined in the Colour field (column) of the layer attribute table. These are previously defined within the ontology library file. Line styles for the connections layer use arrows to denote direction of mass flux/influence, and style to determine the nature of the connection. These style files are designed to aid the mapping process, where large symbols are readily selectable for connecting, editing and grouping. For display and review purposes, where clear labels and connections are important, an alternative set of style files (provided separately) can be used to prioritise the component labels over the symbols and improve the differentiation between connection types.

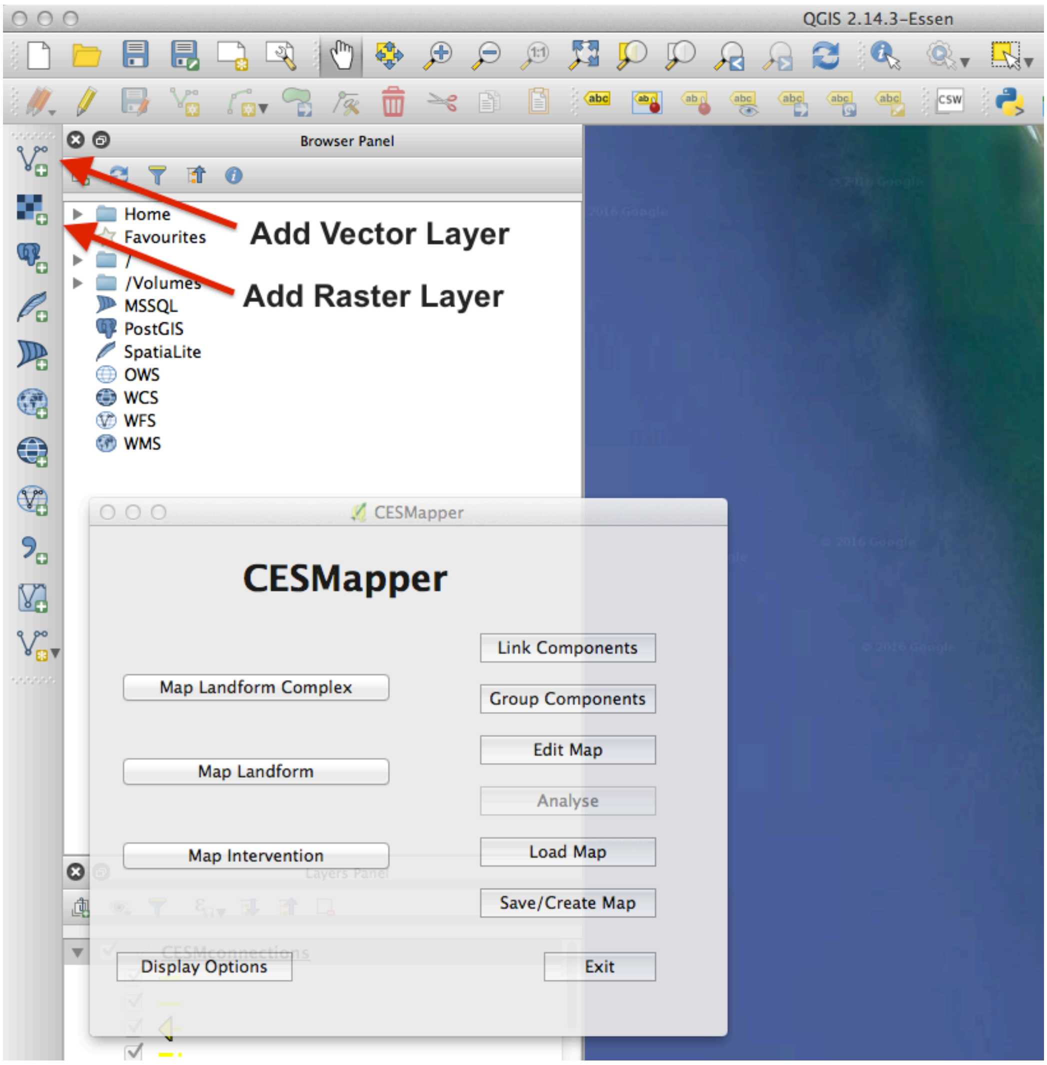


Figure 3.12 Adding additional vector or raster layers within QGIS.

3.10 Manual addition of supplementary information to attribute tables

Slots are provided in the attribute tables for the components and connections. These can be edited within QGIS to include additional information, such as quantitative values for sediment fluxes, or links to meta-data or other sources of information.

4. Frequently Asked Questions

Q: Performance seems slow with a large map – is there anything that can be done to improve the responsiveness within QGIS?
A: Performance can be slow if a large number of QGIS plugins are installed. Try uninstalling any that are not needed.

Q: I can't group landforms into a landform complex
A: This functionality will be completed very soon in a new CESMapper release.

Q: CESMapper is missing some important landform types?
A: The set of landforms and human interventions can be changed to suit a particular regional context by editing the external ontology file. This can be done using any text editor. The format of ontology file is described in Appendix B.

Q: CESMapper asks me to load an ontology – where can I find this?
A: CESMapper releases come with an Ontologies directory. This contains at least one example ontology and should be copied to a convenient location on your computer.

Q: I have found a bug in CESMapper. How do I report it?
A: CESMapper is still at an early release stage. Any can be reported directly to the maintainers (j.french@ucl.ac.uk and h.burningham@ucl.ac.uk). We'll do our best to fix these in a timely fashion.

Q: Where can I get the latest update from?
A: The latest version of CESMapper can be downloaded from the public GitHub repository at: https://github.com/UCL-CERU/CESMapper

References

ABPmer, 2008. Development and demonstration of systems-based estuary simulators. R&D Technical Report FD2117/TR. Defra, London, 241 pp.

Basco, D.R., 2006. Seawall impacts on adjacent beaches: separating fact and fiction. *Journal of Coastal Research* 39, 741-44.

Batten, B.K., Blanton, B., Taylor, S., Plummer, J., 2015. Modeling the influence of sea level rise on future storm surge elevations considering landscape evolution. In: Wang, P., Rosati, J.D., Cheng, J. (Eds.), *Proceedings Coastal Sediments 2015*, World Scientific, 15 pp.

Bowen, A.J., Inman, D.L.,1966. Budget of littoral sands in the vicinity of Point Arguello, California. *US Army Coastal Engineering Research Center Technical Memorandum* 19, 41pp.

Brown, S., Barton, M., Nicholls, R., 2011. Coastal retreat and/or advance adjacent to defences in England and Wales. *Journal of Coastal Conservation* 15, 659-670.

Burgess, K., Jay, H., Hosking, A., 2004. FutureCoast: Predicting the future coastal evolution of England and Wales. *Journal of Coastal Conservation* 10, 65-71.

Cañas, A.J., Carff, R., Hill, G., Carvalho, M., Arguedas, M., Eskridge, T.C., Lott, J., Carvajal, R., 2005. Concept maps: Integrating knowledge and information visualization. In: Tergan, S.O., Keller, T. (Eds.), *Knowledge and information visualization: searching for synergies*. Lecture Notes in Computer Science 3426, pp. 205-219.

Chini, N., Stansby, P., Leake, J., Wolf, J., Roberts-Jones, J., Lowe, J., 2010. The impact of sea level rise and climate change on inshore wave climate: A case study for East Anglia (UK). *Coastal Engineering* 57, 973-984.

Cooper, N.J., Barber, P.C., Bray, M.J., Carter, D.J., 2002. Shoreline management plans: a national review and engineering perspective. *Proceedings of the Institution of Civil Engineers: Water and Maritime Engineering* 154, 221-228.

Cooper, N.J., Pontee, N.I., 2006. Appraisal and evolution of the littoral 'sediment cell' concept in applied coastal management: experiences from England and Wales. *Ocean and Coastal Management* 49, 498-510.

Cowell, P.J., Stive, M.J.F., Niedoroda, A.W., de Vriend, H.J., Swift, D.J.P., Kaminsky, G.M., Capobianco, M., 2003a. The coastal-tract (Part 1): a conceptual approach to aggregated modeling to lower-order coastal change. *Journal of Coastal Research* 19, 812-827.

Cowell, P.J., Stive, M.J.F., Niedoroda, A.W., Swift, D.J.P., de Vriend, H.J., Buijsman, M.C., Nicholls, R.J., Roy, P.S., Kaminsky, G.M., Cleveringa, J., Reed, C.W., de Boer, P.L., 2003b. The coastal-tract (Part 2): applications of aggregated modeling to lower-order coastal change. J*ournal of Coastal Research* 19, 828-848.

Dyer, K.R., Huntley, D.A., 1999. The origin, classification and modelling of sand banks and ridges. *Continental Shelf Research* 18, 1285-1330.

Eliot, I., Nutt, C., Gozzard, B., Higgins, M., Buckley, E., Bowyer, J., 2011. *Coastal Compartments of Western Australia: A Physical Framework for Marine and Coastal Planning*. Report 80-02. Report to the Departments of Environment and Conservation, Planning and Transport. Environmental Protection Authority. Damara WA Pty Ltd., Perth, 76 pp.

French, J.R., Burningham, H., 2009. *Mapping the connectivity of large scale coastal geomorphological systems: Coastal system mapping with CmapTools tutorial*. Science Report – SC060074/PR2. Bristol, Environment Agency, 25pp.

French, J.R., Burningham, H., 2013. Coasts and climate: insights from geomorphology. *Progress in Physical Geography* 37, 550-561.

French, J.R., Burningham, H., Thornhill, G., Whitehouse, R., Nicholls, R.S., 2016a. Conceptualizing and mapping coupled estuary, coast and inner shelf sediment systems. *Geomorphology*. 256, 17-35. http://dx.doi.org/10.1016/j.geomorph.2015.10.006

French, J.R., Payo, A., Murray, A.B., Orford, J., Eliot, M., Cowell, P., 2016b. Appropriate complexity for the prediction of coastal and estuarine geomorphic behaviour at decadal to centennial scales. *Geomorphology* 256, 3-16. http://dx.doi.org/10.1016/j.geomorph.2015.10.005

French, J.R., Burningham, H., Thornhill, G., Nicholls, R.J. 2016c. Integrating estuarine, coastal and inner shelf sediment systems in a common conceptual framework as a basis for participatory shoreline management. In: Meadows, M., Lin, J-C. (eds.) *Geomorphology and society*. Singapore, Springer, 245-277.

Gruber, T.R., 1993. A translation approach to portable ontology specifications. *Knowledge Acquisition* 5, 199-220.

Hall, D.M., Lazarus, E.D., Swannack, T.M., 2014. Strategies for communicating systems models. *Environmental Modelling & Software* 55, 70-76.

Hanson, S., Nicholls, R.J., Balson, P., Brown, I., French, J.R., Spencer, T., Sutherland, W.J., 2010. Capturing coastal geomorphological change within regional integrated assessment: an outcome-driven fuzzy logic approach. *Journal of Coastal Research* 26, 831-842.

Hapke, C.J., Kratzmann, M.G., Himmelstoss, E.A., 2013. Geomorphic and human influence on large-scale coastal change. *Geomorphology* 199, 160-170.

Inman, D.L., Frautschy, J.D., 1966. Littoral processes and the development of shorelines. *Proceedings of the Coastal Engineering Speciality Conference*, Santa Barbara, California. American Society of Civil Engineers, New York, 511-536.

Komar, P.D., 1996. The budget of littoral sediments: concepts and applications. *Shore & Beach* 18-26.

Motyka, J.M., Brampton, A.H., 1993. Coastal Management: Mapping of Littoral Cells. *HR Wallingford Report SR 328*, Hydraulics Research Ltd, Wallingford, UK, 102 pp.

Nicholls, R.J., Townend, I.H., Bradbury, A.P., Ramsbottom, D., Day, S.A., 2013. Planning for long-term coastal change: experience from England and Wales. *Ocean Engineering* 71, 3–16.

Nicholls, R.J., French, J.R., Burnhigham, H., Van Maanen, B., Payo, A., Sutherland, J., Walkden, M., Thornhill, G., Brown, J., Luxford, F., Simm, J., Reeve, D.E., Halll, J.W., Souza, A., Stansby, P.K., Amoudry, L.O., Rogers, B.D., Ellis, M., Whitehouse, R., Horrillo-Caraballo, J.M., Karunarathna, H., Pan, S., Plater, A., Dix, J., Barnes, J., Heron, E. 2015. Improving decadal coastal geomorphic predictions: an overview of the iCOASST project. *Proceedings Coastal Sediments 2015*, San Diego, USA.

Rosati, J.D., 2005. Concepts in sediment budgets. *Journal of Coastal Research* 21, 307-322.

Sanò, M., Richards, R., Medina, R., 2014. A participatory approach for system conceptualization and analysis. *Environmental Modeling & Software* 54, 142-152.

Sayers, P.B., Hall, J.W., Meadowcroft, I.C., 2002. Towards risk-based flood hazard management in the UK. *Proceedings of the Institution of Civil Engineers* 150, 36-42.

Van Maanen, B., Nicholls, R.J., French, J.R., Barkwith, A., Bonaldo, D., Burningham, H., Murray, A.B., Payo, A., Sutherland, J., Thornhill, G., Townend, I.H., van der Wegen, M., Walkden, M.J.A., 2016. Simulating mesoscale coastal evolution for decadal coastal management: A new framework integrating multiple, complementary modelling approaches. *Geomorphology* 256, 68–80. http://dx.doi.org/10.1016/j.geomorph.2015.10.026

Voinov, A., Bousquet, F., 2010. Modelling with stakeholders. *Environmental Modelling & Software* 25, 1268-1281.

Walkden, M., Hall, J., 2011. A mesoscale predictive model of the evolution and management of a soft-rock coast. *Journal of Coastal Research* 27, 529-543.

Whitehouse, R., Balson, P., Beech, N., Brampton, A., Blott, S., Burningham, H., Cooper, N., French, J., Guthrie, G., Hanson, S., Nicholls, R., Pearson, S., Pye, K., Rossington, K., Sutherland, J., Walkden, M., 2009. *Characterisation and prediction of large-scale, long-term change of coastal geomorphological behaviours: Final science report – SC060074/SR1*. Environment Agency, Bristol, 264pp. ISBN: 978-184911-090-7.

Appendix A

License

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2)
tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component
(kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to
control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require,
such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited
permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License,
in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) hat supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded rom the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version,
but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a
later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS
THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

Appendix B

CESM ontology file format

Coastal System Ontology Markup (CSOML) Version 1.4

This document describes a minimal semantic markup language (with optional presentational markup elements) for the specification of coastal system ontologies. The following tags and associated fields are permitted in a CSOML Library Definition file, with those highlighted in **bold** being mandatory (though these may have empty or null entries):

<library> </library>

Define the beginning and end of any Library Definition file: <library> must always be the first tag and </library> must always be the last.

<CSOMLversion> *version* **</CSOMLversion >**

Specify the CSOML version, which is used to check compatibility with the CSM software. Value of *version* must be numeric (integer or real).

<metadata> *Author="name", Date="13/01/2013"* **</metadata>**

Optionally specify additional information, such as author, date, etc. Syntax for metadata is *itemname1="content1",itemname2 = "content2" …* etc.

<libraryname> *libname* **</libraryname>**

Specify the name of the library, *lname* (text string).

<libraryversion> *libversion* </libraryversion>

Optionally, specify the library version. Value of *libversion* must be numeric (integer or real).

<numorders> *m* **</ numorders >**

Specify the number of orders, *m* (integer), in the system hierarchy. Order 0 is a root order that is not actually used by the CSM software, but which is used in graphical representations of the ontology. Order 0 will always be a root order with no sub-orders (see below); the lowest orders will always be landform complexes < landforms < engineering interventions. The minimum value for *m* is therefore 4.

<order0> </order0>
<order1> </order1>
…..
<order*m*> </order*m*>

Define the orders where *n*=0 is lowest order of the hierarchy and *m* the highest. Coastal system ontologies consider order 0 to represent the largest spatial and temporal scales of behavior, with higher orders (up to order *m*) representing smaller scales.

The following tags must reside inside paired **<order*n*>** and **</order*n*>** tags
<ordername> </ordername>

Specify the name of the order. This is used to populate GUI menus in the CSM software.

<property> ***numcomps=n, numsuborders=m****, shape=0, colour=#FFFFFF*

</property>

Specify the properties, including optional presentational markup, for the current order. The following are currently permitted:

numcomps = n Define the number of components in this order (integer *n*).

In the case of an order that contains two or more sub-orders, then n will be the number of sub-orders. If there are no sub-orders, *n* will be the number of types.

numsuborders=m Defines the number of 'sub-orders' (defined below). Optional, if the components are to be separated into groups within a specific order.

shape {0 (default),1}

Defines the mapped shape of the component: 0=rectangle, 1 = circle

colour {=*#FFFFFF* }

Define the mapped colour of the component as a Hex code

<suborder>…. </suborder>

Optionally, specify a sub-order (e.g. Estuary,Open Coast and Inner shelf, for a Landform complex order). Not all orders will necessarily have sub-orders and Order 0 is never allowed sub-orders. If present, there must be at least two sub-orders. Each sub-order must contain all the components that would be otherwise associated with an order.

<subproperty>*subordername='subname', numcomps=k, shape=0, colour==#FFFFFF* </subproperty>

This defines the properties pertaining to the 'suborder', similar to those for the order.

<component> name="compname", *colour=#FFFFFF,* **children="child1,** child2, child3……..childn**" </component>**

Specify each component of current order, with the name "compname", and list the permitted children for that component in children="…." (a quote-terminated and comma-separated list). The colour for the component can be specified as a Hex code, but may be omitted, in which case the order or suborder colour is used. (N.B. if the colour keyword is used, it must appear between the '**name=**' keyword and the '**children=**' keyword.

There is a special case of children="all" where all the components at the next level down are permitted, or children="none" where there are no permitted children.

Notes:

1. A **<component>** ….. **</component>** entry must be followed by a **</ordern>** tag to close the definitions for that order.

2. As noted above, the library definition file must always be terminated by a **</library>** tag.

3. The CSM software performs a check to ensure that the version of CSMOL is

supported: it should always read CSMOL files at or below, but not necessarily above, its current version.

4. Blank lines are permitted and are ignored. Whitespace is ignored.