



&

ETC on Biological Diversity

Nature 2000

Union List and Change Report

Calculation Methodology

Abstract

This document describes the calculation procedures for the production of the Union List and Change Report deliverables published by the ETC on Biological Diversity as part of Natura 2000 indicators.

Ver.	Date	Author	Description	Action*	Pg.
1.0	15/10/2012	Alberto Telletxea	First version	Creation	*
	23/10/2012	Sabine Roscher (ETC/BD)		Minor corrections to the text	

1	INTRODUCTION	3
1.1	Objective and scope.....	3
2	UNION LIST	4
2.1	INPUT DATA USED and OUTPUT STATISTICS PRODUCED	4
2.1.1	Union List input data.....	4
2.1.2	Union List output calculations:	7
2.2	METHODOLOGY and PROCEDURES in USE	8
3	UNION LIST CHANGE REPORT	31
3.1	INPUT DATA USED and OUTPUT STATISTICS PRODUCED	31
3.1.1	Input data:.....	31
3.1.2	Output data:.....	32
3.2	METHODOLOGY and PROCEDURES in USE	33

1 INTRODUCTION

Nature 2000 network data requires periodically a set of reports derived from statistical calculations made on top of the European compilation of the reported data by the MS on Habitats and Birds Directive. This document focuses on the Union List and Change Report deliverables.

1.1 OBJECTIVE AND SCOPE

This document aims to describe the calculation processes executed for obtaining the *Union List* database and *Change Report* for the latest *Union List*. These two deliverables are part of the official Natura2000 network derived products provided to the Commission. Concretely, the *Union List* appears as a table published in the Official Journal of European Communities (<http://eur-lex.europa.eu>).

2 UNION LIST

Union List is the official list and statistics of Habitats Directive SCIs adopted each year by the Habitats Committee and are derived from the descriptive databases reported by Member States. The Union List is presented by bio-geographical regions and Member State.

2.1 INPUT DATA USED AND OUTPUT STATISTICS PRODUCED

2.1.1 Union List input data

The input data for the calculation of the Union List take into account two databases:

1. Most recent consolidated QA/QCed database of Nature 2000. The specifics tables (Sites, Habitats and Species) and fields used are the following:

TABLE	FIELDS
SITES: Register of the Nature 2000 sites	SiteCode: Unique identification code for a Natura2000 site
	SiteName: Name of the site
	SiteType: Type code for the different designations (Habitats or Birds)
	Length: Perimeter of the site in kilometers (not always reported by Member States)
	Area: Area in hectares of a site
TABLE	FIELDS
CONTAINS BIOREGION: Register of the biological regions that contain Natura 2000 sites	SiteCode: Unique identification code for a Natura2000 site
	BiologicalID: Unique identification for the biological region
	BiologicalName: Name for the biological region (Atlantic, Mediterranean, Alpine, Continental, Macaronesian, Boreal, Arctic, Pannonian, Anatolian, Stepic and Black Sea).
TABLE	FIELDS
SPECIES: Register of the species	SiteCode: Unique identification code for a Natura2000 site
	SpeciesName: Latinname of the species

present on Natura 2000 sites	SpeciesGroup: Code for the group, which the species belongs (Amphibian, Fish, Invertebrate, Mammal , Plant)
	Population: Size and density of the population of the species present on the site in relation to the populations present within national territory. • A: 100% $\geq p > 15\%$ • B: 15% $\geq p > 2\%$ • C: 2% $\geq p > 0\%$ • D: non-significant population
	Conservation: Degree of conservation of the features of the habitat which are important for the species concerned and possibilities for restoration. • A: conservation excellent • B: good conservation • C: average or reduced conservation
	GlobalImportance: Global assessment of the value of the site for conservation of the species concerned. • A: excellent value • B: good value • C: significant value
TABLE	FIELDS
HABITATS: Register of the habitat that contain Natura 2000 sites	SiteCode: Unique identification code for a Natura2000 site
	HabitatCode: Identifier for the habitats based in the ETC reference list on habitats
	GlobalAssessment: Global assessment of the value of the site for conservation of the natural habitat type concerned. • A: excellent value • B: good value • C: significant value

	Representativity: Degree of representativity of the natural habitat type on the site. • A: excellent representativity • B: good representativity • C: significant representativity • D: non-significant presence
	Relsurface: Relative surface area of the site covered by the natural habitats type in relation to the total area covered by that natural habitat type within the national territory. • A: $100 \geq p > 15\%$ • B: $15 \geq p > 2\%$ • C: $2 \geq p > 0$
	Consstatus: Degree of conservation status of the structure and functions of the natural habitat type concerned and restoration possibilities • A: excellent conservation • B: good conservation • C: average or reduced conservation

2. The other database in use is an Access mdb file called Priority_features.mdb provided by the ETC. This database consists of the following tables and respective fields:

TABLE	FIELDS
HABITATPRIORITY : Register of the habitats existing in Natura2000 and which of those are designated as priority	Code: Unique identifier for the species
	P_Habitats : Describes which are the priority habitats
	HBTTAXEN : The description name for the habitats

TABLE	FIELDS
SPECIESPRIORITY: Register of the species existing in Natura2000 and which of those are designated as priority	Code: Unique identifier for the species
	P_Species: Describes which are the priority species
	Name: Latinname for the species
	Tax_Group: Code for the group, which the species belongs. • A: Amphibian • F: Fish • I: Invertebrate • M: Mammal • P: Plant

2.1.2 Union List output calculations:

The output calculations produce a list of Natura2000 sites group by MS and bio-geographical region with the following information:

SCIcode	NameSCI	AreaHA	LENGTHKM	Longitude	Latitude	Article1	Criteria
AT1203A00	Ötscher - Dürre	42596,57		15,111111	47,840278	*	1
AT1211A00	Wienerwald -	52168,59		16,125	48,147222	*	1
AT1212A00	Nordöstliche F	64084,57		15,983333	47,883333	*	1

TABLE	FIELDS
UNION LIST	SCI Code: Unique identification code for a Natura2000 site
	Site Name: Name of the SCI site
	Area Ha: Area in hectares of a site
	Length Km: Perimeter of the site in kilometers
	Longitude: Geographic coordinates in decimal degrees
	Latitude: Geographic coordinates in decimal degrees
	Article 1: Presence on the SCI of at least one priority natural habitat type and/or species within the meaning of Article 1 of Directive 92/43/EEC
	Criteria: Records the appropriate criteria that a site meets to qualify as an SCI. The designation procedure will be explained in detail in the methodology section.

2.2 METHODOLOGY AND PROCEDURES IN USE

The methodology has been established by the ETC on Biological Diversity. To implement automatically this methodology a set of SQL procedures are executed on top of a relational database where input Natura2000 data tables reside.

The calculation methodology is applied to assign the value for the field *Criteria*. In principle, the criteria are hierarchical with 1 being highest and 6 the lowest. Thus, for the field *Criteria* only highest values are shown. For example, a site will only be quoted as '1' in spite that the site meets also criteria 2, 3, 4, 5 & 6. The methodology in use to set the right value for the field *Criteria* is described in the following graph:

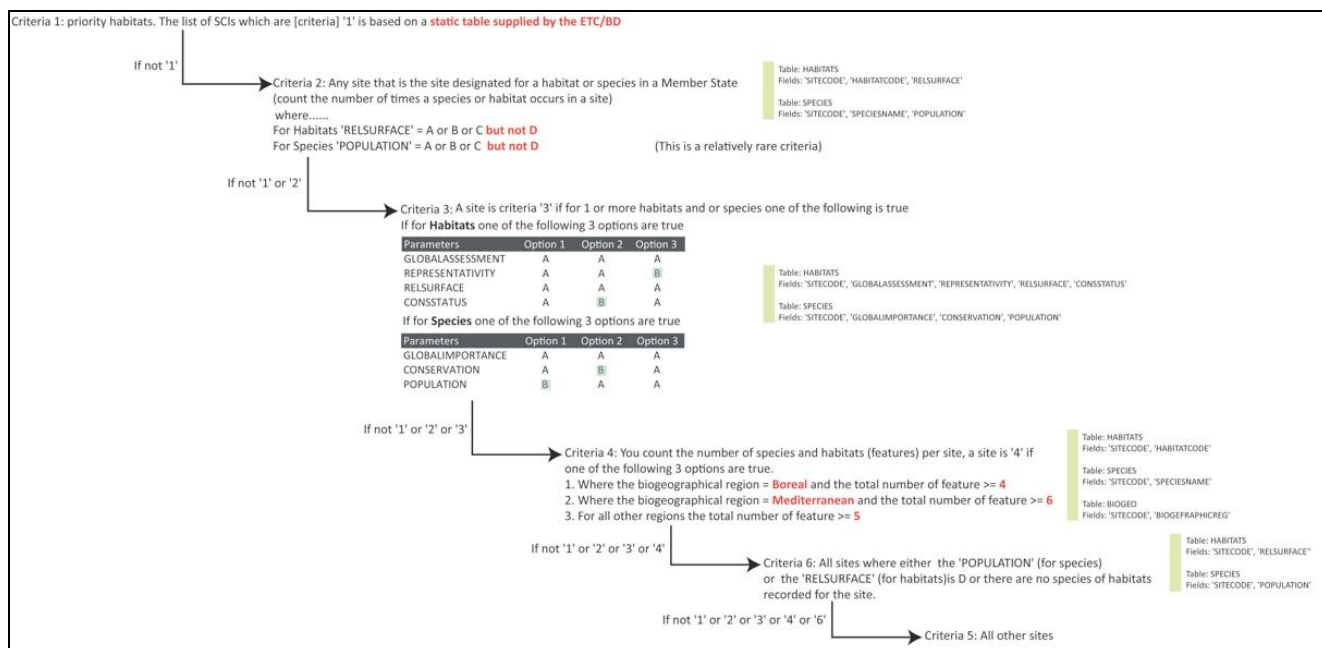


Figure: Overview on methodology to identify the criteria used for selection of sites for the "preliminary Union List
Source: ETC/BD 2011

These are the procedures that automate the designation of the Union List .

1. Import of the ETC tables that resides in the *Priority_Features.mdb*: HABITATPRIORITY and SPECIESPRIORITY

```
SQL
sentence

USE [Natura2KImport]
GO
/***** Object:  StoredProcedure [dbo].[ETC_CommunityListImport]
Script Date: 02/09/2012 17:35:35 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE procedure [dbo].[ETC_CommunityListImport]
as
begin
declare @CommunityListPath nvarchar(128)
Declare @sSQL nvarchar(1000)
set @CommunityListPath = (SELECT [value]FROM [Natura2KImport].[dbo].[Parameters] WHERE parameter
='COMMUNITYLISTPATH')
SET @CommunityListPath = @CommunityListPath +'Priority_features.mdb'
print @CommunityListPath
IF object_id('Natura2000.[dbo].[Habitat_priority]') IS NOT NULL
DROP TABLE [Natura2000].[dbo].[Habitat_priority]
```

```
CREATE TABLE [Natura2000].[dbo].[Habitat_priority](
    [HBCDAX] [nvarchar](4) NULL,
    [HBTAXEN] [nvarchar](254) NULL,
    [P_habitats] [bit] NOT NULL
)
set @sSQL = ' Insert into [Natura2000].[dbo].[Habitat_priority]([HBCDAX],[HBTAXEN],[P_habitats])
SELECT *
FROM OPENROWSET(''Microsoft.Jet.OLEDB.4.0'',
''' + @CommunityListPath + ''';'admin'';'', Habitat_priority)'
Exec(@sSQL)
IF object_id('[Natura2000].[dbo].[Species_priority]') IS NOT NULL
DROP TABLE [Natura2000].[dbo].[Species_priority]

CREATE TABLE [Natura2000].[dbo].[Species_priority](
    [code] [nvarchar](255) NULL,
    [P_species] [bit] NOT NULL,
    [Name] [nvarchar](255) NULL,
    [Tax Group] [nvarchar](255) NULL
)
set @sSQL = ' Insert into [Natura2000].[dbo].[Species_priority]([code],[P_species],[Name],[Tax Group])
SELECT *
FROM OPENROWSET(''Microsoft.Jet.OLEDB.4.0'',
''' + @CommunityListPath + ''';'admin'';'', Species_priority)'
Exec(@sSQL)
IF object_id('[Natura2000].[dbo].[PriorityIF]') IS NOT NULL
DROP TABLE [Natura2000].[dbo].[PriorityIF]

CREATE TABLE [Natura2000].[dbo].[PriorityIF](
    [ID] [bit] NULL,
    [Sitecode] [nvarchar](254) NULL,
    [HabitatCode] [nvarchar](254) NULL
)
set @sSQL = ' Insert into [Natura2000].[dbo].[PriorityIF]([ID],[Sitecode],[HabitatCode])
SELECT *
FROM OPENROWSET(''Microsoft.Jet.OLEDB.4.0'',
''' + @CommunityListPath + ''';'admin'';'', PriorityIF)'
Exec(@sSQL)
end
```

2. Set the value for the field *Criteria*:

SQL
sentence

```
USE [Natura2KImport]
GO
/***** Object: StoredProcedure [dbo].[All_Criteria_List]    Script Date: 02/09/2012 17:35:18 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
```

```
GO

--Create a procedute to sort the Criterias of every Sitecode
CREATE procedure [dbo].[All_Criteria_List]
as

begin
-- CREATE CONTAINSSPECIESNOBIRD: This table has the same information of constainsspecies
except the birds information

CREATE TABLE [Natura2000].[dbo].[CONTAINSSPECIESNOBIRD](
    [SITECODE] [nvarchar](9) NOT NULL,
    [SPECIESNAME] [nvarchar](61) NOT NULL,
    [SPECIESCODE] [nvarchar](4) NULL,
    [POPULATION] [nvarchar](14) NULL,
    [CONSERVATION] [nvarchar](1) NULL,
    [ISOLATIONFACTOR] [nvarchar](1) NULL,
    [GLOBALIMPORTANCE] [nvarchar](1) NULL,
    [RESIDENT] [nvarchar](14) NULL,
    [BREEDING] [nvarchar](14) NULL,
    [WINTER] [nvarchar](14) NULL,
    [STAGING] [nvarchar](14) NULL,
    [MOTIVATION] [nvarchar](1) NULL,
    [STARTDATE] [datetime] NULL,
    [ENDDATE] [datetime] NULL,
    [RID] [int] NOT NULL,
    [OTHERSPECIES] [smallint] NULL,
    [SENSITIVE] [smallint] NULL,
    [SPECIESNAMECLEAN] [nvarchar](61) NULL
)

-- Insert data in [Natura2000].[dbo].[CONTAINSSPECIESNOBIRD]
insert into [Natura2000].[dbo].[CONTAINSSPECIESNOBIRD]([SITECODE] ,[SPECIESNAME],
[SPECIESCODE],[POPULATION],[CONSERVATION],[ISOLATIONFACTOR],[GLOBALIMPORTANCE],
[RESIDENT],[BREEDING],[WINTER],[STAGING],[MOTIVATION],
[STARTDATE],[ENDDATE],[RID],[OTHERSPECIES],[SENSITIVE],[SPECIESNAMECLEAN])
SELECT [SITECODE]
,[SPECIESNAME]
,[SPECIESCODE]
,[POPULATION]
,[CONSERVATION]
,[ISOLATIONFACTOR]
,[GLOBALIMPORTANCE]
,[RESIDENT]
,[BREEDING]
,[WINTER]
,[STAGING]
,[MOTIVATION]
```

```
,[STARTDATE]
,[ENDDATE]
,[RID]
,[OTHERSPECIES]
,[SENSITIVE]
,[SPECIESNAMECLEAN]
FROM [Natura2000].[dbo].[CONTAINSSPECIES]
where SPECIESNAME in (SELECT LATINNAME
FROM [Natura2000].[dbo].[SPECIES]
where SPTYPE not in('B'))

-- These temporal tables are used to generate Criteria2

CREATE TABLE [Natura2000].[dbo].[Criterio2bis](
    [SITECODE] [nvarchar](9) NULL
)
CREATE TABLE [Natura2000].[dbo].[Criterio2](
    [SITECODE] [nvarchar](9) NULL
)
CREATE TABLE [Natura2000].[dbo].[no_Criterio2](
    [SITECODE] [nvarchar](9) NULL
)
--Table MS_SCI: it has a list with all sitecode
CREATE TABLE [Natura2000].[dbo].[MS_SCI](
    [SITECODE] [nvarchar](9) NULL,
    [COUNTRY_CO] [nvarchar](4) NULL
)
-- insert data in the table MS_SCI
Insert into [Natura2000].[dbo].[MS_SCI]([SITECODE],[COUNTRY_CO] )
SELECT [SITECODE]
,[COUNTRY_CODE]
FROM [Natura2000].[dbo].[NATURASITE]
where SITETYPE in ('B','C','E','G','I','K')

-- Table Yes_Habitat_priority: it has a priority list by habitats
CREATE TABLE [Natura2000].[dbo].[Yes_Habitat_priority](
    [HBCDAX] [nvarchar](4) NULL,
    [P_habitats] [bit] NOT NULL,
    [HBTAXEN] [nvarchar](254) NULL
)
-- insert data in the table Yes_Habitat_Priority
insert into [Natura2000].[dbo].[Yes_Habitat_priority]([HBCDAX],[P_habitats],[HBTAXEN])
SELECT [HBCDAX]
,[P_habitats]
,[HBTAXEN]
```

```

FROM [Natura2000].[dbo].[Habitat_priority]
where p_habitats = 1

-- Table Yes_Species_priority: it has a priority list by species
CREATE TABLE [Natura2000].[dbo].[Yes_Species_priority](
    [CODE] [nvarchar](254) NULL,
    [P_SPECIES] [bit] NOT NULL,
    [NAME] [nvarchar](254) NULL,
    [TAX_GROUP] [nvarchar](254) NULL
)

-- insert data in the table Yes_Species_priority
insert into [Natura2000].[dbo].[Yes_Species_priority]([CODE] ,[P_SPECIES],[NAME],[TAX_GROUP])
    SELECT[code]
        ,[P_species]
        ,[Name]
        ,[Tax Group]
    FROM [Natura2000].[dbo].[Species_priority]
    where p_species =1

-- Table Criterio1_habitat
CREATE TABLE [Natura2000].[dbo].[Criterio1_habitat](
    [SITECODE] [nvarchar](9) NULL,
)

-- insert data in the table Criterio1_habitat
insert into [Natura2000].[dbo].[Criterio1_habitat]([SITECODE])
    SELECT MS_SCI.SITECODE
    FROM ([Natura2000].[dbo].MS_SCI INNER JOIN [Natura2000].[dbo].CONTAINSHABITAT
    ON [Natura2000].[dbo].ms_SCI.SITECODE =
[Natura2000].[dbo].CONTAINSHABITAT.SITECODE) INNER JOIN
[Natura2000].[dbo].Yes_Habitat_priority ON
[Natura2000].[dbo].CONTAINSHABITAT.HABITATCODE = [Natura2000].[dbo].yes_Habitat_priority.HBCDAX
    GROUP BY MS_SCI.SITECODE;

-- Table Criterio1_Species
CREATE TABLE [Natura2000].[dbo].[Criterio1_Species](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table Criterio1_Species
insert into [Natura2000].[dbo].[Criterio1_Species]([SITECODE])
    SELECT MS_SCI.SITECODE
    FROM ([Natura2000].[dbo].MS_SCI INNER JOIN
[Natura2000].[dbo].CONTAINSSPECIESNOBIRD ON MS_SCI.SITECODE
= CONTAINSSPECIESNOBIRD.SITECODE) INNER JOIN
[Natura2000].[dbo].Yes_Species_priority
ON CONTAINSSPECIESNOBIRD.SPECIESCODE = Yes_Species_priority.CODE

```

```

GROUP BY MS_SCI.SITECODE;

-- Table Criterio1 : SiteCode by Criteria1
CREATE TABLE [Natura2000].[dbo].[Criterio1](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table Criterio1
insert into [Natura2000].[dbo].[Criterio1]([SITECODE])
    SELECT MS_SCI.SITECODE
    FROM ([Natura2000].[dbo].MS_SCI
INNER JOIN [Natura2000].[dbo].CONTAINSHABITAT
ON ms_SCI.SITECODE = CONTAINSHABITAT.SITECODE) INNER
JOIN [Natura2000].[dbo].Yes_Habitat_priority ON
CONTAINSHABITAT.HABITATCODE = yes_Habitat_priority.HBCDAX

GROUP BY MS_SCI.SITECODE

-- insert data in the table Criterio1
insert into [Natura2000].[dbo].[Criterio1]([SITECODE])
SELECT [SITECODE]
    FROM [Natura2000].[dbo].[Criterio1_Species]
    where SITECODE not in (select SITECODE
from [Natura2000].[dbo].Criterio1_habitat)

-- Table no_Criterio1:SiteCode dont belong to Criteria1
CREATE TABLE [Natura2000].[dbo].[no_Criterio1](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table
insert into [Natura2000].[dbo].[no_Criterio1]([SITECODE])
SELECT [SITECODE]
    FROM [Natura2000].[dbo].[MS_SCI]
    where SITECODE not in (select SITECODE
                                From [Natura2000].[dbo].Criterio1)

/* CRITERIO 2 */

declare @country nvarchar(2)
declare @country1 nvarchar(20)

DECLARE Countries CURSOR FOR
    SELECT [country]
    FROM [Natura2000].[dbo].[countries]
    --where country ='fr'

```

```

order by country

OPEN Countries

FETCH NEXT FROM Countries INTO @country

WHILE @@FETCH_STATUS = 0
BEGIN

    set @country1 = @country+'%'
    print @country1
    --Table Criteria2_Habitat
    CREATE TABLE [Natura2000].[dbo].[Criteria2_Habitat](
        [SITECODE] [nvarchar](9) NULL,
        [HABITATCODE] [nvarchar](4) NULL
    )

    -- insert data in the table Criteria2_Habitat
    Insert into [Natura2000].[dbo].[Criteria2_Habitat] ([SITECODE],[HABITATCODE] )
        SELECT MS_SCI.SITECODE, CONTAINSHABITAT.HABITATCODE
        FROM [Natura2000].[dbo].MS_SCI
INNER JOIN [Natura2000].[dbo].CONTAINSHABITAT ON MS_SCI.SITECODE = CONTAINSHABITAT.SITECODE
        Where [Natura2000].[dbo].MS_SCI.SITECODE
like + @country1 ----*****

    -- Table criteria2_habitat_count_equals_1: list with the number
of times a habitat occurs in a site
    CREATE TABLE [Natura2000].[dbo].[criteria2_habitat_count_equals_1](
        [HABITATCODE] [nvarchar](4) NULL,
        [CountOfHABITATCODE] [int] NULL
    )

    -- insert data in the table criteria2_habitat_count_equals_1
    Insert into [Natura2000].[dbo].[criteria2_habitat_count_equals_1]
    ([HABITATCODE],[CountOfHABITATCODE] )
        SELECT CONTAINSHABITAT.HABITATCODE,
Count(CONTAINSHABITAT.HABITATCODE) AS CountOfHABITATCODE
        FROM [Natura2000].[dbo].CONTAINSHABITAT
        where [Natura2000].[dbo].CONTAINSHABITAT.SITECODE
like + @country1 ----*****

        GROUP BY CONTAINSHABITAT.HABITATCODE
        HAVING (((Count(CONTAINSHABITAT.HABITATCODE))=1));

    -- Table LV_criteria_2_habitat_count_equal_1_SITE
    CREATE TABLE [Natura2000].[dbo].[LV_criteria_2_habitat_count_equal_1_SITE](
        [SITECODE] [nvarchar](9) NULL,
        [HABITATCODE] [nvarchar](4) NULL
    )

    -- insert data in the table LV_criteria_2_habitat_count_equal_1_SITE
    INSERT INTO [Natura2000].[dbo].

```

```
[LV_criteria_2_habitat_count_equal_1_SITE]([SITECODE],[HABITATCODE] )

        SELECT MS_SCI.SITECODE, criteria2_habitat_count_equals_1.HABITATCODE
        FROM
        [Natura2000].[dbo].criteria2_habitat_count_equals_1 INNER JOIN
        ([Natura2000].[dbo].CONTAINSHABITAT
INNER JOIN [Natura2000].[dbo].MS_SCI
        ON CONTAINSHABITAT.SITECODE = MS_SCI.SITECODE)
        ON criteria2_habitat_count_equals_1.HABITATCODE =
CONTAINSHABITAT.HABITATCODE
where [Natura2000].[dbo].CONTAINSHABITAT.SITECODE like + @country1 ---*****

-- Criterio2 Species

-- Table Criteria2_Species
CREATE TABLE [Natura2000].[dbo].[Criteria2_Species](
        [SITECODE] [nvarchar](9) NULL,
        [SPECIESCODE] [nvarchar](8) NULL
)

-- insert data in the table Criteria2_Species
Insert into [Natura2000].[dbo].[Criteria2_Species] ([SITECODE],[SPECIESCODE] )
        SELECT MS_SCI.SITECODE, CONTAINSSPECIESNOBIRD.SPECIESCODE
        FROM [Natura2000].[dbo].
MS_SCI INNER JOIN [Natura2000].[dbo].CONTAINSSPECIESNOBIRD ON MS_SCI.SITECODE =
CONTAINSSPECIESNOBIRD.SITECODE

Where [Natura2000].[dbo].MS_SCI.sitecode like + @country1 ---*****

-- Table Criteria2_species_count_equals_1
CREATE TABLE [Natura2000].[dbo].[Criteria2_species_count_equals_1](
        [SPECIESCODE] [nvarchar](8) NULL
)

-- insert data in the table Criteria2_species_count_equals_1

insert into [Natura2000].[dbo].[Criteria2_species_count_equals_1]([SPECIESCODE])

SELECT CONTAINSSPECIESNOBIRD.SPECIESCODE

FROM [Natura2000].[dbo].CONTAINSSPECIESNOBIRD

where [Natura2000].[dbo].CONTAINSSPECIESNOBIRD.SITECODE like + @country1 ---*****

GROUP BY CONTAINSSPECIESNOBIRD.SPECIESCODE

HAVING (((Count(CONTAINSSPECIESNOBIRD.SPECIESCODE))=1));
```

```
-- Table LV_Criteria2_species_count_equal_2_SITE

CREATE TABLE [Natura2000].[dbo].[LV_Criteria2_species_count_equal_2_SITE](
    [SITECODE] [nvarchar](9) NULL,
    [SPECIESCODE] [nvarchar](8) NULL
)

-- insert data in the table LV_Criteria2_species_count_equal_2_SITE

Insert into [Natura2000].[dbo].[LV_Criteria2_species_count_equal_2_SITE]([SITECODE] ,[SPECIESCODE])
SELECT MS_SCI.SITECODE, Criteria2_species_count_equals_1.SPECIESCODE
FROM
    [Natura2000].[dbo].MS_SCI INNER JOIN
        ([Natura2000].[dbo].CONTAINSSPECIESNOBIRD
INNER JOIN [Natura2000].[dbo].Criteria2_species_count_equals_1
ON CONTAINSSPECIESNOBIRD.SPECIESCODE = Criteria2_species_count_equals_1.SPECIESCODE)
ON MS_SCI.SITECODE = CONTAINSSPECIESNOBIRD.SITECODE
where [Natura2000].[dbo].CONTAINSSPECIESNOBIRD.SITECODE like + @country1 ----*****

-- Table Criteria2_Insert
CREATE TABLE [Natura2000].[dbo].[Criteria2_Insert](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table Criteria2_Insert
Insert into [Natura2000].[dbo].[Criteria2_Insert]([SITECODE])
select    [Natura2000].[dbo].no_Criterio1.SITECODE
FROM ([Natura2000].[dbo].no_Criterio1 inner join [Natura2000].[dbo].CONTAINSHABITAT
on no_Criterio1.SITECODE=CONTAINSHABITAT.SITECODE) inner join
[Natura2000].[dbo].LV_criteria_2_habitat_count_equal_1_SITE on CONTAINSHABITAT.HABITATCODE
=criteria_2_habitat_count_equal_1_SITE.HABITATCODE
where RELSURFACE in ('A','B','C')
and [Natura2000].[dbo].no_Criterio1.SITECODE like + @country1

-- insertar species y habitats sitecode
Insert into [Natura2000].[dbo].[Criteria2_Insert]([SITECODE])
select    [Natura2000].[dbo].no_Criterio1.SITECODE FROM
([Natura2000].[dbo].no_Criterio1 inner join [Natura2000].[dbo].CONTAINSSPECIESNOBIRD
on no_Criterio1.SITECODE=CONTAINSSPECIESNOBIRD.SITECODE)
inner join [Natura2000].[dbo].LV_Criteria2_species_count_equal_2_SITE
on CONTAINSSPECIESNOBIRD.SPECIESCODE
= Criteria2_species_count_equal_2_SITE.SPECIESCODE
where POPULATION in ('A','B','C')
and [Natura2000].[dbo].no_Criterio1.SITECODE like + @country1
```

```

insert into [Natura2000].[dbo].[Criterio2bis]([SITECODE])
SELECT [SITECODE]
FROM [Natura2000].[dbo].[Criteria2_Insert]

Drop Table [Natura2000].[dbo].[Criteria2_Habitat]
Drop Table [Natura2000].[dbo].[criteria2_habitat_count_equals_1]
Drop Table [Natura2000].[dbo].[LV_criteria_2_habitat_count_equal_1_SITE]
Drop Table [Natura2000].[dbo].[Criteria2_Species]
Drop Table [Natura2000].[dbo].[Criteria2_species_count_equals_1]
Drop Table [Natura2000].[dbo].[LV_Criteria2_species_count_equal_2_SITE]
Drop Table [Natura2000].[dbo].[Criteria2_Insert]

FETCH NEXT FROM Countries INTO @country
END
CLOSE countries
DEALLOCATE countries

-- Insert SiteCodes in [Natura2000].[dbo].[Criterio2]
insert into [Natura2000].[dbo].[Criterio2]([SITECODE])
SELECT distinct [SITECODE]
FROM [Natura2000].[dbo].[Criterio2bis]

-- Insert SiteCodes in [Natura2000].[dbo].[no_Criterio2]
not belong to Criteria 1 and 2
insert into [Natura2000].[dbo].[no_Criterio2]([SITECODE])
SELECT [SITECODE]
FROM [Natura2000].[dbo].[no_Criterio1]
where sitecode not in ( SELECT [SITECODE]
FROM [Natura2000].[dbo].[Criterio2])

-- ***** criterio 3

--Create table [Natura2000].[dbo].[Species_v1]
CREATE TABLE [Natura2000].[dbo].[Species_v1](
    [SITECODE] [nvarchar](9) NULL,
    [SPECIESNAME] [nvarchar](61) NULL,
    [SPECIESCODE] [nvarchar](8) NULL,
    [POPULATION] [nvarchar](14) NULL,
    [CONSERVATION] [nvarchar](1) NULL,
    [ISOLATIONFACTOR] [nvarchar](1) NULL,
    [GLOBALIMPORTANCE] [nvarchar](1) NULL,
    [SPTYPE] [nvarchar](1) NULL
)

-- insert data in the table [Natura2000].[dbo].[Species_v1]

```

```

insert into [Natura2000].[dbo].[Species_v1]([SITECODE] ,[SPECIESNAME] ,
[SPECIESCODE] ,[POPULATION] ,
[CONSERVATION] ,[ISOLATIONFACTOR] ,[GLOBALIMPORTANCE] ,[SPTYPE])
SELECT DISTINCT C.[SITECODE]
,[SPECIESNAME]
,s.[SPECIESCODE]
,[POPULATION]
,[CONSERVATION]
,[ISOLATIONFACTOR]
,[GLOBALIMPORTANCE]
,[SPTYPE]
FROM [Natura2000].[dbo].[SPECIES] S
,[Natura2000].[dbo].[CONTAINSSPECIESNOBIRD] C
,[Natura2000].[dbo].[NATURASITE] NS
where S.LATINNAMECLEAN = C.SPECIESNAME
AND NS.SITECODE = C.SITECODE
AND sptype not in('B')
and CONSERVATION is not null
and GLOBALIMPORTANCE is not null
and POPULATION is not null
and SITETYPE in ('B','C','E','G','I','K')

-- Table Criterio3_INSERT
CREATE TABLE [Natura2000].[dbo].[Criterio3_INSERT](
[SITECODE] [nvarchar](9) NULL
)

-- insert data in the table Criterio3_INSERT
Insert into [Natura2000].[dbo].[Criterio3_INSERT]([SITECODE])
SELECT no_Criterio2.SITECODE
FROM
[Natura2000].[dbo].no_Criterio2 INNER JOIN [Natura2000].[dbo].[Species_v1]
ON no_Criterio2.SITECODE = [Species_v1].SITECODE
WHERE
((([Species_v1].GLOBALIMPORTANCE)='A')
AND (([Species_v1].CONSERVATION)='A')
AND (([Species_v1].POPULATION)='B'))
OR
((([Species_v1].GLOBALIMPORTANCE)='A')
AND (([Species_v1].CONSERVATION)='B')
AND (([Species_v1].POPULATION)='A'))
OR
((([Species_v1].GLOBALIMPORTANCE)='A')
AND (([Species_v1].CONSERVATION)='A')
AND (([Species_v1].POPULATION)='A')));

-- insert data in the table Criterio3_INSERT
insert into [Natura2000].[dbo].[Criterio3_INSERT]([SITECODE])

```

```

SELECT no_Criterio2.SITECODE
FROM [Natura2000].[dbo].no_Criterio2 INNER JOIN [Natura2000].[dbo].CONTAINSHABITAT
ON no_Criterio2.SITECODE = CONTAINSHABITAT.SITECODE
WHERE (((CONTAINSHABITAT.GLOBALASSESSMENT)='A')
AND ((CONTAINSHABITAT.REPRESENTATIVITY)='A')
AND ((CONTAINSHABITAT.RELSURFACE)='A')
AND ((CONTAINSHABITAT.CONSSTATUS)='A'))
OR
(((CONTAINSHABITAT.GLOBALASSESSMENT)='A')
AND ((CONTAINSHABITAT.REPRESENTATIVITY)='A')
AND ((CONTAINSHABITAT.RELSURFACE)='A')
AND ((CONTAINSHABITAT.CONSSTATUS)='B'))
OR
(((CONTAINSHABITAT.GLOBALASSESSMENT)='A')
AND ((CONTAINSHABITAT.REPRESENTATIVITY)='B')
AND ((CONTAINSHABITAT.RELSURFACE)='A')
AND ((CONTAINSHABITAT.CONSSTATUS)='A')));

-- Table Criterio3
CREATE TABLE [Natura2000].[dbo].[Criterio3](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table Criterio3
insert into [Natura2000].[dbo].[Criterio3]([SITECODE])
SELECT [SITECODE]
FROM [Natura2000].[dbo].[no_Criterio2]
where sitecode in (SELECT [SITECODE]
FROM [Natura2000].[dbo].[Criterio3_INSERT])

CREATE TABLE [Natura2000].[dbo].[no_Criterio3](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table
insert into [Natura2000].[dbo].[no_Criterio3]([SITECODE])
SELECT [SITECODE]
FROM [Natura2000].[dbo].[no_Criterio2]
where sitecode not in (SELECT [SITECODE]
FROM [Natura2000].[dbo].[Criterio3])
/*end
*/

-- criterio 4 *****

CREATE TABLE [Natura2000].[dbo].[criteria4_hab_count](
    [SITECODE] [nvarchar](9) NULL,
    [HabCount] [int] NULL

```

```

)

-- insert data in the table Criteria4_hab_count
insert into [Natura2000].[dbo].[criteria4_hab_count]([SITECODE],[HabCount])
SELECT no_Criterio3.SITECODE,
Count(CONTAINSHABITAT.HABITATCODE) AS CountOfHABITATCODE
FROM
[Natura2000].[dbo].no_Criterio3 INNER JOIN [Natura2000].[dbo].CONTAINSHABITAT
ON no_Criterio3.SITECODE = CONTAINSHABITAT.SITECODE
GROUP BY no_Criterio3.SITECODE;

-- Table criteria4_Spec_count
CREATE TABLE [Natura2000].[dbo].[criteria4_Spec_count](
    [SITECODE] [nvarchar](9) NULL,
    [SpecCount] [int] NULL
)

-- insert data in the table criteria4_Spec_count
insert into [Natura2000].[dbo].[criteria4_Spec_count]([SITECODE],[SpecCount])
SELECT no_Criterio3.SITECODE,
Count(CONTAINSSPECIESNOBIRD.SPECIESCODE) AS CountOfSPECIESCODE
FROM
[Natura2000].[dbo].no_Criterio3,[Natura2000].[dbo].CONTAINSSPECIESNOBIRD,
[Natura2000].[dbo].SPECIES,[Natura2000].[dbo].NATURASITE
where no_Criterio3.SITECODE = CONTAINSSPECIESNOBIRD.SITECODE
and SPECIES.LATINNAME=CONTAINSSPECIESNOBIRD.SPECIESNAME
and NATURASITE.SITECODE=CONTAINSSPECIESNOBIRD.SITECODE
and sptype not in('B')
and SITETYPE in ('B','C','E','G','I','K')
GROUP BY no_Criterio3.SITECODE;

-- Table criteria4_step2_a
CREATE TABLE [Natura2000].[dbo].[criteria4_step2_a](
    [SITECODE] [nvarchar](9) NULL,
    [Hab] [int] NULL,
    [Spec] [int] NULL
)

-- insert data in the table criteria4_step2_a
insert into [Natura2000].[dbo].[criteria4_step2_a]([SITECODE],[Hab],[Spec] )
SELECT no_Criterio3.SITECODE, criteria4_hab_count.HabCount,
criteria4_Spec_count.SpecCount
FROM [Natura2000].[dbo].criteria4_Spec_count RIGHT JOIN
([Natura2000].[dbo].criteria4_hab_count RIGHT JOIN [Natura2000].[dbo].no_Criterio3
ON criteria4_hab_count.SITECODE = no_Criterio3.SITECODE)
ON criteria4_Spec_count.SITECODE = no_Criterio3.SITECODE;

Update [Natura2000].[dbo].[criteria4_step2_a]

```

```

set Hab=0
where Hab is null

Update [Natura2000].[dbo].[criteria4_step2_a]
set Spec=0
where Spec is null

----

CREATE TABLE [Natura2000].[dbo].[Criterio4_INSERT](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table Criterio4_INSERT
insert into [Natura2000].[dbo].[Criterio4_INSERT]([SITECODE] )
    SELECT distinct [Natura2000].[dbo].[criteria4_step2_a].[SITECODE]
    FROM [Natura2000].[dbo].[criteria4_step2_a]
    , [Natura2000].[dbo].[BIOGEOGRAPHICREG]
    , [Natura2000].[dbo].[BELONGSTOBIOREGION]
    where
[Natura2000].[dbo].[BELONGSTOBIOREGION].SITECODE=[Natura2000].[dbo].
[criteria4_step2_a].SITECODE

and [Natura2000].[dbo].[BELONGSTOBIOREGION].BIOREGID=[Natura2000].[dbo].
[BIOGEOGRAPHICREG].ID
    and
    ((DESCRIPTION='Boreal' and ([Hab]+[Spec]) >=4)
    or
    (DESCRIPTION='Mediterranean' and ([Hab]+[Spec]) >=6)
    or
    ([Hab]+[Spec]) >=5 and (DESCRIPTION not in('Boreal','Mediterranean'))))

-- Table Criterio4
CREATE TABLE [Natura2000].[dbo].[Criterio4](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table Criterio4
insert into [Natura2000].[dbo].[Criterio4]([SITECODE])
    SELECT [SITECODE]
    FROM [Natura2000].[dbo].[no_Criterio3]
    where sitecode in (SELECT [SITECODE]
    FROM [Natura2000].[dbo].[Criterio4_INSERT])

--Table no_Criterio4
CREATE TABLE [Natura2000].[dbo].[no_Criterio4](
    [SITECODE] [nvarchar](9) NULL
)

```

```
-- insert data in the table no_Criterio4
insert into [Natura2000].[dbo].[no_Criterio4]([SITECODE] )
SELECT [SITECODE]
FROM [Natura2000].[dbo].[no_Criterio3]
where sitecode not in (SELECT [SITECODE]
FROM [Natura2000].[dbo].[Criterio4])

-- ***** criterio 6

-- Create table [Natura2000].[dbo].[testcriteria6]
CREATE TABLE [Natura2000].[dbo].[testcriteria6](
[SITECODE] [nvarchar](9) NULL)

declare @SitecodeID nvarchar(9)
declare @SpeciesName nvarchar(61)
declare @habitatCode nvarchar(4)
declare @cont int
set @cont=0

DECLARE Sitecode CURSOR FOR
SELECT distinct SITECODE
FROM [Natura2000].[dbo].no_Criterio4
order by SITECODE

-- Open cursor SiteCode to take the sitecode not belong to Criteria 1,2,3 and 4
OPEN Sitecode
FETCH NEXT FROM Sitecode INTO @SitecodeID
WHILE @@FETCH_STATUS = 0
BEGIN
print @SitecodeID

-- ***** Cursor species

DECLARE Species CURSOR FOR
SELECT SPECIESNAME
FROM [Natura2000].[dbo].CONTAINSSPECIESNOBIRD
Where sitecode=@SitecodeID

OPEN Species
FETCH NEXT FROM Species INTO @SpeciesName
WHILE @@FETCH_STATUS = 0
BEGIN
print @SpeciesName
```

```

        if (not exists(SELECT sitecode, [SPECIESNAME],POPULATION
                        FROM [Natura2000].[dbo].[CONTAINSSPECIESNOBIRD]
                        where SITECODE=+@SitecodeID and SPECIESNAME=+@SpeciesName
                        and (POPULATION = ('D') or POPULATION is null)))
            set @cont=1

        FETCH NEXT FROM Species INTO @SpeciesName

    END
    CLOSE Species
    DEALLOCATE Species

-- ***** end cursor species

-- ***** Cursor habitat
    DECLARE habitat CURSOR FOR
    SELECT HABITATCODE
        FROM [Natura2000].[dbo].CONTAINSHABITAT
        Where sitecode=+@SitecodeID

    OPEN habitat
    FETCH NEXT FROM habitat INTO @habitatCode
    WHILE @@FETCH_STATUS = 0
    BEGIN
        print @habitatCode

        if (not exists(SELECT sitecode, HABITATCODE
                        FROM [Natura2000].[dbo].CONTAINSHABITAT
                        where SITECODE=+@SitecodeID and HABITATCODE=+@habitatCode
                        and (RELSURFACE = ('D') or RELSURFACE is null)))
            set @cont=1

        FETCH NEXT FROM habitat INTO @habitatCode

    END
    CLOSE habitat
    DEALLOCATE habitat
-- ***** cursor end habitat

    if @cont=0
        insert into Natura2000.dbo.testcriteria6(SITECODE) Values (@SitecodeID)
    else
        set @cont=0

    FETCH NEXT FROM Sitecode INTO @SitecodeID
    END
    CLOSE Sitecode

```

```

DEALLOCATE Sitecode

-- Create Table [Natura2000].[dbo].[Criterio6_noHabitat]
CREATE TABLE [Natura2000].[dbo].[Criterio6_noHabitat](
    [SITECODE] [nvarchar](9) NULL
)

-- Insert data [Natura2000].[dbo].[Criterio6_noHabitat] Table
Insert into [Natura2000].[dbo].[Criterio6_noHabitat]([SITECODE])
    SELECT NATURASITE.SITECODE FROM [Natura2000].[dbo].
NATURASITE LEFT JOIN [Natura2000].[dbo].CONTAINSHABITAT
    ON naturasite.SITECODE = CONTAINSHABITAT.SITECODE
where CONTAINSHABITAT.SITECODE is null

-- Create table [Natura2000].[dbo].[Criterio6_noSpecies]
CREATE TABLE [Natura2000].[dbo].[Criterio6_noSpecies](
    [SITECODE] [nvarchar](9) NULL
)

-- Insert data [Natura2000].[dbo].[Criterio6_noSpecies] table
Insert into [Natura2000].[dbo].[Criterio6_noSpecies]([SITECODE])
    SELECT NATURASITE.SITECODE FROM [Natura2000].[dbo].
NATURASITE LEFT JOIN [Natura2000].[dbo].CONTAINSSPECIESNOBIRD
    ON naturasite.SITECODE = CONTAINSSPECIESNOBIRD.SITECODE
where CONTAINSSPECIESNOBIRD.SITECODE is null

-- Insert data Natura2000.dbo.testcriteria6 table
insert into Natura2000.dbo.testcriteria6(SITECODE)
    select Criterio6_noSpecies.SITECODE
    from [Natura2000].[dbo].Criterio6_noHabitat inner join
[Natura2000].[dbo].Criterio6_noSpecies
    on Criterio6_noHabitat.SITECODE = Criterio6_noSpecies.SITECODE
    where Criterio6_noSpecies.SITECODE in(select SITECODE
    from [Natura2000].[dbo].no_Criterio4)

-- Create table [Natura2000].[dbo].[Criterio6]
CREATE TABLE [Natura2000].[dbo].[Criterio6](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table Criterio6
Insert into [Natura2000].[dbo].[Criterio6]([SITECODE])
    select distinct sitecode
    from Natura2000.dbo.testcriteria6

-- CREATE TABLE [Natura2000].[dbo].[no_Criterio6]

```

```

CREATE TABLE [Natura2000].[dbo].[no_Criterio6](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table no_Criterio6
insert into [Natura2000].[dbo].[no_Criterio6]([SITECODE] )
SELECT [SITECODE]
FROM [Natura2000].[dbo].[no_Criterio4]
where sitecode not in (SELECT [SITECODE]
FROM [Natura2000].[dbo].[Criterio6])
-- ***** criterio 5

-- Table Criterio5
CREATE TABLE [Natura2000].[dbo].[Criterio5](
    [SITECODE] [nvarchar](9) NULL
)

-- insert data in the table Criterio5
insert into [Natura2000].[dbo].[Criterio5]([SITECODE])
SELECT [SITECODE]
FROM [Natura2000].[dbo].[no_Criterio6]

CREATE TABLE [Natura2000].[dbo].[Sitecode_Criteria1](
    [SITECODE] [nvarchar](9) NULL)

Insert into [Natura2000].[dbo].[Sitecode_Criteria1]([SITECODE])
SELECT [SITECODE]
FROM [Natura2000].[dbo].[NATURASITE]

CREATE TABLE [Natura2000].[dbo].[Sitecode_Criteria1_temp](
    [SITECODE] [nvarchar](9) NULL)

Insert into [Natura2000].[dbo].[Sitecode_Criteria1_temp]([SITECODE])
SELECT distinct [SITECODE]
FROM [Natura2000].[dbo].[Sitecode_Criteria1]
where SITECODE not in (SELECT [SITECODE]
FROM [Natura2000].[dbo].[Criterio1])

Insert into [Natura2000].[dbo].[Criterio1]([SITECODE])
SELECT distinct [SITECODE]
FROM [Natura2000].[dbo].[Sitecode_Criteria1]
where SITECODE not in (SELECT [SITECODE]
FROM [Natura2000].[dbo].[Criterio1])

Delete from [Natura2000].[dbo].[Criterio2]

```

```

        where SITECODE in (SELECT [SITECODE]
        FROM [Natura2000].[dbo].[Sitecode_Criteria1_temp])

Delete from [Natura2000].[dbo].[Criterio3]
        where SITECODE in (SELECT [SITECODE]
        FROM [Natura2000].[dbo].[Sitecode_Criteria1_temp])

Delete from [Natura2000].[dbo].[Criterio4]
        where SITECODE in (SELECT [SITECODE]
        FROM [Natura2000].[dbo].[Sitecode_Criteria1_temp])

Delete from [Natura2000].[dbo].[Criterio5]
        where SITECODE in (SELECT [SITECODE]
        FROM [Natura2000].[dbo].[Sitecode_Criteria1_temp])

Delete from [Natura2000].[dbo].[Criterio6]
        where SITECODE in (SELECT [SITECODE]
        FROM [Natura2000].[dbo].[Sitecode_Criteria1_temp])

-- Delete all temporal tables

--Drop Table [Natura2000].[dbo].Sitecode_Criteria1
Drop Table [Natura2000].[dbo].Sitecode_Criteria1_temp
--Drop Table [Natura2000].[dbo].Criteria2_Habitat
--Drop Table [Natura2000].[dbo].criteria2_habitat_count_equals_1
--Drop Table [Natura2000].[dbo].Criteria2_Insert
--Drop Table [Natura2000].[dbo].Criteria2_Species
--Drop Table [Natura2000].[dbo].Criteria2_species_count_equals_1
Drop Table [Natura2000].[dbo].Criterio2bis
Drop Table [Natura2000].[dbo].criteria4_hab_count
Drop Table [Natura2000].[dbo].criteria4_Spec_count
Drop Table [Natura2000].[dbo].criteria4_step2_a
Drop Table [Natura2000].[dbo].Criterio1_habitat
Drop Table [Natura2000].[dbo].Criterio1_Species
Drop Table [Natura2000].[dbo].Criterio3_INSERT
Drop Table [Natura2000].[dbo].Criterio4_INSERT
Drop Table [Natura2000].[dbo].MS_SCI
Drop Table [Natura2000].[dbo].no_Criterio1
Drop Table [Natura2000].[dbo].no_Criterio2
Drop Table [Natura2000].[dbo].no_Criterio3
Drop Table [Natura2000].[dbo].no_Criterio4
Drop Table [Natura2000].[dbo].no_Criterio6
Drop Table [Natura2000].[dbo].Species_v1
Drop Table [Natura2000].[dbo].Yes_Habitat_priority
Drop Table [Natura2000].[dbo].Yes_Species_priority
Drop Table [Natura2000].[dbo].testcriteria6
Drop Table [Natura2000].[dbo].criterio6_nohabitat
Drop Table [Natura2000].[dbo].criterio6_nospecies

```

	Drop Table [Natura2000].[dbo].CONTAINSSPECIESNOBIRD end
--	--

3. Create a final Union List table group by biogeographical region and MS.

SQL sentence	<pre> USE [Natura2KImport] GO /***** Object: StoredProcedure [dbo].[Biogeographic_MS_List] Script Date: 02/09/2012 17:35:22 *****/ SET ANSI_NULLS ON GO SET QUOTED_IDENTIFIER ON GO CREATE procedure [dbo].[Biogeographic_MS_List] as begin declare @Release nvarchar(128) set @Release = (select top 1 [value] from [Natura2000].[dbo].info_METADATA where [parameter] = 'Database Version') declare @country_code nvarchar(4) declare @DESCRIPTION nvarchar(255) declare @nameTable nvarchar(40) declare @DataBase nvarchar(255) set @DataBase = '[n2kmanager].[dbo].'</pre> <pre> -- ***** Cursor ***** DECLARE UnionList_MSBioRegion CURSOR FOR Select distinct [COUNTRY_CODE] as country_code from [Natura2000].[dbo].[NATURASITE] order by [COUNTRY_CODE] OPEN UnionList_MSBioRegion FETCH NEXT FROM UnionList_MSBioRegion INTO @country_code WHILE @@FETCH_STATUS = 0 BEGIN DECLARE UnionList_BioRegionMS CURSOR FOR </pre>
-----------------	---

```

select
DISTINCT DESCRIPTION
from
[Natura2000].[dbo].NATURASITE as A, [Natura2000].[dbo].BELONGSTOBIOREGION As b,
[Natura2000].[dbo].BIOGEGRAPHICREG as c
where
A.SITECODE = b.SITECODE
and b.BIOREGID = c.ID
and A.SITETYPE in
('C','B','E','G','I','K')
and A.COUNTRY_CODE =
@country_code
OPEN UnionList_BioRegionMS
FETCH NEXT FROM UnionList_BioRegionMS INTO @DESCRIPTION
WHILE @@FETCH_STATUS = 0
BEGIN
    if @DESCRIPTION='Black Sea'
    begin
        Set @DESCRIPTION='BlackSea'
        set @nameTable=@country_code+'_'+@DESCRIPTION+'_UnionList_'+@Release
        Set @DESCRIPTION='Black Sea'
    end
    else
        set @nameTable=@country_code+'_'+@DESCRIPTION+'_UnionList_'+@Release
    IF object_id(@DataBase + @nameTable ) IS NOT NULL
        EXECUTE('DROP TABLE ' + @DataBase + @nameTable);
    print @nameTable
    EXECUTE('CREATE TABLE ' + @DataBase + @nameTable + '(
        SCICode [nvarchar](9) NULL,
        NameSCI [nvarchar](240) NULL,
        --Article1 [nvarchar] (1) NULL,
        AreaHA [numeric](38, 4) NULL,
        LENGTHKM [numeric](38, 4) NULL,
        Longitude [numeric](38, 6) NULL ,
        Latitude [numeric](38, 6) NULL,
        --Criteria int,
    ));
    EXECUTE('insert into ' + @DataBase + @nameTable + '(SCICode,NameSCI,AreaHA,LENGTHKM,Longitude,Latitude)
    SELECT distinct
        A.[SITECODE]
        ,[SITENAME]
        ,[AREAHA]
        ,[LENGTHKM]
        ,[LONGITUDE]

```

```

                                ,[LATITUDE]

FROM

                                [Natura2000].[dbo].NATURASITE as A,
                                [Natura2000].[dbo].BELONGSTOBIOREGION As b,
                                [Natura2000].[dbo].BIOGEOGRAPHICREG as c

Where

                                A.SITECODE = b.SITECODE
                                and b.BIOREGID = c.ID
                                and A.SITETYPE in ('C','B','E','G','I','K')
                                and COUNTRY_CODE= '' + @country_code + ''
                                and DESCRIPTION= '' + @DESCRIPTION + '';

EXECUTE('ALTER TABLE ' + @DataBase + @nameTable + '
        ADD Article1 [nvarchar] (1) NULL');

EXECUTE('ALTER TABLE ' + @DataBase + @nameTable + '
        ADD Criteria [int] NULL');

EXECUTE('UPDATE ' + @DataBase + @nameTable + '
        SET Criteria = 1, Article1= ''*''
        FROM [Natura2000].[dbo].[Criterio1] JOIN [n2kmanager].[dbo]. ' + @nameTable + '
        ON [Natura2000].[dbo].[Criterio1].SITECODE = [n2kmanager].[dbo]. ' + @nameTable + '
        '.SCICode Collate SQL_Latin1_General_CP1_CI_AI');

EXECUTE('UPDATE ' + @DataBase + @nameTable + '
        SET Criteria = 2
        FROM [Natura2000].[dbo].[Criterio2] JOIN [n2kmanager].[dbo]. ' + @nameTable + '
        ON [Natura2000].[dbo].[Criterio2].SITECODE = [n2kmanager].[dbo]. ' + @nameTable + '
        '.SCICode Collate SQL_Latin1_General_CP1_CI_AI');

EXECUTE('UPDATE ' + @DataBase + @nameTable + '
        SET Criteria = 3
        FROM [Natura2000].[dbo].[Criterio3] JOIN [n2kmanager].[dbo]. ' + @nameTable + '
        ON [Natura2000].[dbo].[Criterio3].SITECODE = [n2kmanager].[dbo]. ' + @nameTable + '
        '.SCICode Collate SQL_Latin1_General_CP1_CI_AI');

EXECUTE('UPDATE ' + @DataBase + @nameTable + '
        SET Criteria = 4
        FROM [Natura2000].[dbo].[Criterio4] JOIN [n2kmanager].[dbo]. ' + @nameTable + '
        ON [Natura2000].[dbo].[Criterio4].SITECODE = [n2kmanager].[dbo]. ' + @nameTable + '
        '.SCICode Collate SQL_Latin1_General_CP1_CI_AI');

EXECUTE('UPDATE ' + @DataBase + @nameTable + '
        SET Criteria = 5
        FROM [Natura2000].[dbo].[Criterio5] JOIN [n2kmanager].[dbo]. ' + @nameTable + '

```

```

ON [Natura2000].[dbo].[Criterio5].SITECODE = [n2kmanager].[dbo].'+ @nameTable +
'.SCICode Collate SQL_Latin1_General_CP1_CI_AI');

EXECUTE('UPDATE '+ @DataBase + @nameTable +
SET Criteria = 6
FROM [Natura2000].[dbo].[Criterio6] JOIN [n2kmanager].[dbo].'+ @nameTable + '
ON [Natura2000].[dbo].[Criterio6].SITECODE = [n2kmanager].[dbo].'+ @nameTable +
'.SCICode Collate SQL_Latin1_General_CP1_CI_AI');

FETCH NEXT FROM UnionList_BioRegionMS INTO @DESCRIPTION

END

CLOSE UnionList_BioRegionMS

DEALLOCATE UnionList_BioRegionMS

FETCH NEXT FROM UnionList_MSBioRegion INTO @country_code

End

```

3 UNION LIST CHANGE REPORT

The change report will allow the comparison of the Natura 2000 Union Lists of two releases (current and previous).

3.1 INPUT DATA USED AND OUTPUT STATISTICS PRODUCED

3.1.1 Input data:

The input data are two Union List (preliminary Union list from the current year and the Union List from the previous year) tables with the following fields:

TABLE	FIELDS
UNION LIST	SCICode: Unique identification code for a Natura2000 site
	SiteName: Name of the SCI site
	AreaHa: Area in hectares of a site
	LengthKm: Perimeter of the site in kilometers
	Longitude: Geographical coordinates in decimal degrees
	Latitude: Geographical coordinates in decimal degrees
	Article1: Presence on the SCI of at least one priority natural habitat type and/or

	species within the meaning of Article 1 of Directive 92/43/EEC
	Criteria: Records the appropriate criteria that a site meets to qualify as an SCI. The designation procedure will be explained in detail in the methodology section.

3.1.2 Output data:

The output data is a database that contains different tables to reflect the following calculations:

1. Change of sites from one Union List to the other:
 - a. Sites added in the latest Union List from the previous
 - b. Sites deleted in the latest Union List from the previous
2. Change of values from one Union List to the other:
 - a. Latitude
 - b. Longitude
 - c. Area
 - d. Length
 - e. Criteria

These are the tables and fields provided by the calculations:

TABLE	FIELDS
SITESADDED	SCICode: Unique identification code for a Natura2000 site
	SiteName: Name of the SCI site
	AreaHa: Area in hectares of a site
	LengthKm: Perimeter of the site in kilometers
	Longitude: Geographical coordinates in decimal degrees
	Latitude: Geographical coordinates in decimal degrees

TABLE	FIELDS
SITESDELETED	SCICode: Unique identification code for a Natura2000 site
	SiteName: Name of the SCI site
	AreaHa: Area in hectares of a site

	LengthKm: Perimeter of the site in kilometers
	Longitude: Geographic coordinates in decimal degrees
	Latitude: Geographic coordinates in decimal degrees

TABLE	FIELDS
SITESCHANGE	SCICode: Unique identification code for a Natura2000 site
	SiteName: Name of the SCI site
	Change: In area, longitude, latitude, length and criteria
	OldValue: Value from the previous Union List
	NewValue: Value from the current Union List
	AreaPercentageDiffers: The percentage difference in area from one to the other Union List

3.2 METHODOLOGY AND PROCEDURES IN USE

The methodology has been established by the ETC on Biological Diversity. To implement automatically this methodology a set of SQL procedures are executed on top of a relational database where input Natura 2000 data resides.

These are the methodology steps in use and the related procedures for the automation.

1. Calculate the sites added and sites deleted by comparing one each other:

SQL sentence	<pre> USE [Natura2KImport] GO /***** Object: StoredProcedure [dbo].[Change_of_Sites] Script Date: 02/09/2012 17:38:08 *****/ SET ANSI_NULLS ON GO SET QUOTED_IDENTIFIER ON GO CREATE procedure [dbo].[Change_of_Sites] --@rele nvarchar(9),@relePath nvarchar(255),@rele2 nvarchar(9),@relePath2 nvarchar(255) as </pre>
--------------	---

```

begin

IF object_id('n2kmanager.[dbo].[SitesByRegion]') IS NOT NULL
    DROP TABLE n2kmanager.[dbo].[SitesByRegion]

CREATE TABLE n2kmanager.[dbo].[SitesByRegion](
    [SITECODE] [nvarchar](max) NOT NULL,
    [DESCRIPTION] [nvarchar](max) NULL
)
Insert into n2kmanager.[dbo].[SitesByRegion]([SITECODE] ,[DESCRIPTION] )
select BE.SITECODE,BIO.DESCRPTION
from [Natura2000].dbo.BELONGSTOBIOREGION as BE,
     [Natura2000].dbo.BIOGEOGRAPHICREG as BIO
Where BE.BIOREGID = BIO.ID

-- Get the previous release name
-- *****

declare @Release1 nvarchar(128)
declare @Release2 nvarchar(128)
set @Release1 ='End2010'
set @Release2 ='End2011'

print @Release
print @Release1
print @Release2

IF object_id('n2kmanager.dbo.'+'@Release1+'_SiteCode' ) IS NOT NULL
Begin
    -- Create SitesAdded Table with the information of the news sitecodes
    IF object_id('n2kmanager.[dbo].[SitesAdded]') IS NOT NULL
        DROP TABLE n2kmanager.[dbo].[SitesAdded]

    CREATE TABLE n2kmanager.[dbo].[SitesAdded](
        [SITECODE] [nvarchar](9) NOT NULL,
        [SITENAME] [nvarchar](240) NULL,
        [AREAHA] [numeric](38, 4) NULL,
        --[LENGTHKM] [numeric](38, 2) NULL,
        --[LATITUDE] [numeric](38, 6) NULL,
        --[LONGITUDE] [numeric](38, 6) NULL
    )

    Execute('Insert into n2kmanager.[dbo].[SitesAdded]([SITECODE] ,[SITENAME],[AREAHA] )
            SELECT  n2kmanager.[dbo].'+@Release+'_SiteCode.[SITECODE]
                    ,[SITENAME]
                    ,[AREAHA]

```

```

FROM n2kmanager.[dbo].'+@Release+'_SiteCode
Where n2kmanager.[dbo].'+@Release+'_SiteCode.SITECODE not in
(SELECT [SITECODE]
FROM n2kmanager.[dbo].'+@Release1+'_SiteCode');

ALTER TABLE n2kmanager.dbo.SitesAdded
ADD [Region] [nvarchar](max) NULL
ALTER TABLE n2kmanager.dbo.SitesAdded
ADD [MS] [nvarchar](2) NULL

-- Create SitesDeleted Table with the information of the sitecodes have been deleted
IF object_id('n2kmanager.[dbo].[SitesDeleted]') IS NOT NULL
DROP TABLE n2kmanager.[dbo].[SitesDeleted]

CREATE TABLE n2kmanager.[dbo].[SitesDeleted](
[SITECODE] [nvarchar](9) NOT NULL,
[SITENAME] [nvarchar](240) NULL,
[AREAHA] [numeric](38, 4) NULL,
--[LENGTHKM] [numeric](38, 2) NULL,
--[LATITUDE] [numeric](38, 6) NULL,
--[LONGITUDE] [numeric](38, 6) NULL
)

Execute(' Insert into n2kmanager.[dbo].[SitesDeleted]([SITECODE] ,[SITENAME],[AREAHA] )
SELECT n2kmanager.[dbo].'+@Release1+'_SiteCode.[SITECODE]
,[SITENAME]
,[AREAHA]
FROM n2kmanager.[dbo].'+@Release1+'_SiteCode
Where n2kmanager.[dbo].'+ @Release1 +'_SiteCode.SITECODE not in
(SELECT [SITECODE]
FROM n2kmanager.[dbo].'+@Release +'_SiteCode)
');

ALTER TABLE n2kmanager.dbo.SitesDeleted
ADD [Region] [nvarchar](max) NULL
ALTER TABLE n2kmanager.dbo.SitesDeleted
ADD [MS] [nvarchar](2) NULL

Declare @code nvarchar(9)
Declare @region nvarchar(240)
Declare @pais nvarchar(9)

DECLARE Codigo CURSOR FOR
SELECT SITECODE,DESCRIPTION
FROM n2kmanager.[dbo].SitesByRegion

```

```

print'=====
OPEN Codigo

        FETCH NEXT FROM Codigo INTO @code,@Region
        WHILE @@FETCH_STATUS = 0
            BEGIN
                print @code
                print @Region

                set @pais = SUBSTRING(@code, 1, 2)

                EXECUTE('UPDATE n2kmanager.dbo.SitesAdded
                        SET Region='' +@Region+
                        ''' WHERE SiteCode='' +@code+'''' );

                EXECUTE('UPDATE n2kmanager.dbo.SitesAdded
                        SET MS='' +@pais+
                        ''' WHERE SiteCode='' +@code+'''' );

                EXECUTE('UPDATE n2kmanager.dbo.SitesDeleted
                        SET Region='' +@Region+
                        ''' WHERE SiteCode='' +@code+'''' );

                EXECUTE('UPDATE n2kmanager.dbo.SitesDeleted
                        SET MS='' +@pais+
                        ''' WHERE SiteCode='' +@code+'''' );

                FETCH NEXT FROM Codigo INTO @code, @Region
            END
        CLOSE Codigo
        DEALLOCATE Codigo

end
else
        Print 'We have not the previous release'

--*-----

-- Sites modified

-- Create a BiogeographicByMS
-- *****
IF object_id('n2kmanager.[dbo].[BiogeographicalByMS]') IS NOT NULL
        DROP TABLE n2kmanager.[dbo].[BiogeographicalByMS]

        CREATE TABLE n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME] [nvarchar](240) NULL)

```

```

INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('at_Alpine');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('at_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('be_Atlantic');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('be_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('bg_Alpine');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('bg_BlackSea');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('bg_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('cy_Mediterranean');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('cz_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('cz_Pannonian');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('de_Alpine');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('de_Atlantic');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('de_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('dk_Atlantic');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('dk_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('ee_Boreal');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('es_Alpine');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('es_Atlantic');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('es_Macaronesian');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('es_Mediterranean');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('fi_Alpine');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('fi_Boreal');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('fr_Alpine');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('fr_Atlantic');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('fr_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('fr_Mediterranean');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('gr_Mediterranean');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('hu_Pannonian');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('ie_Atlantic');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('it_Alpine');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('it_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('it_Mediterranean');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('lt_Boreal');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('lu_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('lv_Boreal');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('mt_Mediterranean');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('nl_Atlantic');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('pl_Alpine');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('pl_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('pt_Atlantic');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('pt_Macaronesian');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('pt_Mediterranean');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('ro_Alpine');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('ro_BlackSea');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('ro_Continental');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('ro_Pannonian');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('ro_Stepic');
INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('se_Alpine');

```

	<pre> INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('se_Boreal'); INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('se_Continental'); INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('si_Alpine'); INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('si_Continental'); INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('sk_Alpine'); INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('sk_Pannonian'); INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('uk_Atlantic'); INSERT INTO n2kmanager.[dbo].[BiogeographicalByMS]([SITENAME])values ('uk_Mediterranean'); end </pre>
--	---

2. Calculate the change of values: This calculation takes into account the differences in area, longitude, latitude, length and criteria by comparing all these fields simultaneously in just one procedure and writing in a table the changes found.

SQL sentence	<pre> USE [Natura2KImport] GO /***** Object: StoredProcedure [dbo].[Change_of_Values] Script Date: 02/09/2012 17:38:09 *****/ SET ANSI_NULLS ON GO SET QUOTED_IDENTIFIER ON GO CREATE procedure [dbo].[Change_of_Values] --@rele nvarchar(9),@relePath nvarchar(255),@rele2 nvarchar(9),@relePath2 nvarchar(255) as begin declare @site nvarchar(125) declare @DESCRIPTION nvarchar(255) declare @Release nvarchar(128) set @Release='End2010' declare @Release1 nvarchar(128) set @Release1='End2011' declare @ReleaseName1 nvarchar(128) declare @ReleaseName2 nvarchar(128) declare @nameTable nvarchar(40) declare @Release2 nvarchar(128) declare @ReleaseStart nvarchar(128) declare @ReleaseEnd Numeric declare @Check int declare @Cont int IF object_id('n2kmanager.dbo.Sites_Change_Criteria') IS NOT NULL EXECUTE('DROP TABLE </pre>
--------------	--

```
n2kmanager.dbo.Sites_Change_Criteria');

EXECUTE('CREATE TABLE n2kmanager.dbo.Sites_Change_Criteria(
    [SiteCode] [nvarchar](9) NULL,
    '+@Release+' int NULL,
    '+@Release1+' int NULL
)');

DECLARE SitesChange_Criteria CURSOR FOR
    SELECT [SITENAME]
    FROM n2kmanager.[dbo].[BiogeographicalByMS]
    --where Sitename not like 'at%'
    --and Sitename not like 'cy%'
    Order by Sitename

OPEN SitesChange_Criteria
    FETCH NEXT FROM SitesChange_Criteria INTO @site
    WHILE @@FETCH_STATUS = 0
        BEGIN
            print 'n2kmanager.dbo.Sites_Change_' + @site
            -- *****

            IF object_id('n2kmanager.dbo.Sites_Change_' + @site ) IS NOT NULL
                EXECUTE('DROP TABLE
n2kmanager.dbo.Sites_Change_' + @site);

            EXECUTE('CREATE TABLE n2kmanager.dbo.Sites_Change_' + @site + '(
                [SiteCode] [nvarchar](9) NULL,
                [SiteName] [nvarchar](240) NULL,
                [Change] [nvarchar](240) NULL,
                [Old_Value] [numeric](38, 4) not NULL,
                [New_Value] [numeric](38, 4) not NULL
            )');

            --[Area_differs] [nvarchar](255) NULL,
            --[Coordinate_differs] [nvarchar](255) NULL

            /*
            EXECUTE('insert into n2kmanager.dbo.Sites_Change_' + @site + '
(SiteCode,SiteName,Change,End2009,Mid2010)
                values(null,null,null,null,null)');

            */

            Set @ReleaseName1 = @site + '_UnionList_' + @Release
            Set @ReleaseName2 = @site + '_UnionList_' + @Release1
            print @ReleaseName1
            print @ReleaseName2
```

```

--- Area --

        print 'Area'

        EXECUTE('insert into n2kmanager.dbo.Sites_Change_' + @site + '
(SiteCode,SiteName,Change,Old_Value,New_Value)

        SELECT R1.[SCICode]
                ,R1.[NameSCI]
                ,R2.[NameSCI]
                ,R2.AreaHA
                ,R1.AreaHA

as R1        FROM [n2kmanager].[dbo].' + @ReleaseName1 + '
                ,[n2kmanager].[dbo].' + @ReleaseName2 + ' as R2
WHERE
        R1.SCICode = R2.SCICode
        and R1.AreaHA != R2.AreaHA'

        );

        print 'update area'

        EXECUTE('UPDATE n2kmanager.dbo.Sites_Change_' + @site + '
                SET Change = ''Area''');

        EXECUTE('ALTER TABLE n2kmanager.dbo.Sites_Change_' + @site + '
                ADD [Area_differs_percentages] [nvarchar](255) NULL');

        EXECUTE('ALTER TABLE n2kmanager.dbo.Sites_Change_' + @site + '
                ADD [Region] [nvarchar](max) NULL');

Declare @code nvarchar(9)
Declare @region nvarchar(240)
Declare @pais nvarchar(9)

DECLARE Codigo CURSOR FOR
        SELECT SITECODE,DESCRIPTION
        FROM n2kmanager.[dbo].SitesByRegion

print'===== '
OPEN Codigo

        FETCH NEXT FROM Codigo INTO @code,@Region
        WHILE @@FETCH_STATUS = 0
        BEGIN
                print @code
                print @Region

                set @pais = SUBSTRING(@code, 1, 2)

        EXECUTE('UPDATE n2kmanager.dbo.Sites_Change_' + @site + '

```

```

SET Region='' +@Region+
    '' WHERE SiteCode='' +@code+'' );

    FETCH NEXT FROM Codigo INTO @code, @Region

END

CLOSE Codigo
DEALLOCATE Codigo

--EXECUTE('ALTER TABLE n2kmanager.dbo.Sites_Change_' + @site + '
--
    ADD [Coordinate_differs] [nvarchar](255) NULL');

_*****
    IF object_id('n2kmanager.dbo.Sites_Change_Temporal' ) IS NOT NULL
        EXECUTE('DROP TABLE n2kmanager.dbo.Sites_Change_Temporal');

    EXECUTE('CREATE TABLE n2kmanager.dbo.Sites_Change_Temporal(
        [SiteCode] [nvarchar](9) NULL );');

    EXECUTE('insert into n2kmanager.dbo.Sites_Change_Temporal (SiteCode)
    Select SiteCode
    FROM n2kmanager.dbo.Sites_Change_' + @site + '
    ');

    if ((select count(*) from n2kmanager.dbo.Sites_Change_Temporal) = 0 )
        exec('DROP TABLE n2kmanager.dbo.Sites_Change_' + @site);

Declare @old numeric(38,4)
Declare @new numeric(38,4)
--Declare @code nvarchar(9)
Declare @Change nvarchar(240)
Declare @result numeric(38,4)

execute('DECLARE Differ CURSOR FOR
    SELECT SiteCode,Change,Old_value,New_value
    FROM n2kmanager.dbo.Sites_Change_' + @site);

print @site
print'=====
OPEN Differ

    FETCH NEXT FROM Differ INTO @code,@Change,@old,@new
    WHILE @@FETCH_STATUS = 0
        BEGIN

            print 'old'
            print @old

```

```

print 'new'
print @new
if @Change='Area'
begin
    if @new > @old
    begin
        set @result=((@new - @old) / @new) * 100
        print @result
        print @code
        print @site
        if @result >5
        begin
            EXECUTE('UPDATE n2kmanager.dbo.Sites_Change_'
+ @site + '
                                SET
Area_differs_percentages = '+@result+
                                'WHERE
SiteCode='''+@code+'''' and Change=''Area'' '
                                );
        end
    end
    else
    begin
        set @result=((@old - @new) / @old) * 100
        print @result
        print @code
        print @site
        if @result >5
        begin
            EXECUTE('UPDATE n2kmanager.dbo.Sites_Change_'
+ @site + '
                                SET
Area_differs_percentages = '+@result+
                                'WHERE
SiteCode='''+@code+'''' and Change=''Area'' '
                                );
        end
    end
end
--else

--if @Change='Longitude'
--begin
--    if @new > @old
--    begin
--        print '=====
--        print '=====
--        print @site
--        print @code
--        print @new

```

```
--
--      print @old
--
--      set @result = @new - @old
--
--      print @result
--
--      if @result >0.04
--
--      begin
--
--          EXECUTE('UPDATE  n2kmanager.dbo.Sites_Change_'
+ @site + '
--
--          SET
Coordinate_differs = '+@result+
--
--          'WHERE
SiteCode=''+'@code+''' and Change='Longitude'' '
--
--          );
--
--      end
--
--      end
--else
--
--      begin
--
--          print '=====
--          print '=====
--
--          print @site
--
--          print @code
--
--          print @new
--
--          print @old
--
--          set @result = @old - @new
--
--          print @result
--
--          if @result >0.04
--
--          begin
--
--          EXECUTE('UPDATE  n2kmanager.dbo.Sites_Change_'
+ @site + '
--
--          SET
Coordinate_differs = '+@result+
--
--          'WHERE
SiteCode=''+'@code+''' and Change='Longitude'' '
--
--          );
--
--      end
--
--      end
--end
--else if @Change='Latitude'
--begin
--
--      if @new > @old
--
--      begin
--
--          print '=====
--          print '=====
--
--          print @site
--
--          print @code
--
--          print @new
--
--          print @old
--
--          set @result = @new - @old
--
--          print @result
--
--          if @result >0.04
--
--          begin
```

```

+ @site + '                                --                                EXECUTE('UPDATE  n2kmanager.dbo.Sites_Change_'
--
Coordinate_differs = '+@result+                                SET
--
SiteCode=''+'@code+'' and Change='Latitude''                                'WHERE
--
--                                );
--
--                                end
--                                end
--                                else
--                                begin
--                                print '=====
--                                print '=====
--                                print @site
--                                print @code
--                                print @new
--                                print @old
--                                set @result = @old - @new
--                                print @result
--                                if @result >0.04
--                                begin
+ @site + '                                --                                EXECUTE('UPDATE  n2kmanager.dbo.Sites_Change_'
--
Coordinate_differs = '+@result+                                SET
--
SiteCode=''+'@code+'' and Change='Latitude'' '                                'WHERE
--
--                                );
--
--                                end
--                                end
--end

FETCH NEXT FROM Differ INTO @code,@Change,@old,@new

END
CLOSE Differ
DEALLOCATE Differ

-- *****
print 'Insert Sites Criteria' + @site
IF object_id('n2kmanager.dbo.Sites_Change_Criteria_' + @site ) IS NOT NULL
EXECUTE('DROP TABLE
n2kmanager.dbo.Sites_Change_Criteria_' + @site);

EXECUTE('CREATE TABLE  n2kmanager.dbo.Sites_Change_Criteria_' +
@site + '(
[SiteCode] [nvarchar](9) NULL,
[SiteName] [nvarchar](240) NULL,
'+@Release+' int NULL,

```

```

'+@Release1+' int NULL,
[Reason] [nvarchar] (255) NULL
)');

EXECUTE('insert into n2kmanager.dbo.Sites_Change_Criteria_' + @site + '
(SiteCode,SiteName,'+@Release+', '+@Release1+')

SELECT R1.[SCICode]
,R1.[NameSCI]
,R1.Criteria
,R2.Criteria
FROM n2kmanager.[dbo].'+ @ReleaseName1 +' as
R1
,n2kmanager.[dbo].'+ @ReleaseName2 +' as R2
WHERE
R1.SCICode = R2.SCICode
and R1.Criteria != R2.Criteria');

EXECUTE('UPDATE n2kmanager.dbo.Sites_Change_Criteria_' + @site + '
SET Reason = Sentence
From n2kmanager.dbo.Reason
where n2kmanager.dbo.Sites_Change_Criteria_' + @site + '
and n2kmanager.dbo.Sites_Change_Criteria_' + @site + '
.END2010=Reason.Old
.END2011=Reason.New');

EXECUTE('ALTER TABLE n2kmanager.dbo.Sites_Change_Criteria_' + @site + '
ADD [Region] [nvarchar](max) NULL');

-- Declare @code nvarchar(9)
--Declare @region nvarchar(240)
--Declare @pais nvarchar(9)

DECLARE Codigo CURSOR FOR
SELECT SITECODE,DESCRIPTION
FROM n2kmanager.[dbo].SitesByRegion

print'=====
OPEN Codigo

FETCH NEXT FROM Codigo INTO @code,@Region
WHILE @@FETCH_STATUS = 0
BEGIN
print @code
print @Region

```

```

set @pais = SUBSTRING(@code, 1, 2)

EXECUTE('UPDATE n2kmanager.dbo.Sites_Change_Criteria_' + @site + '
        SET Region='' + @Region+
        '' WHERE SiteCode='' + @code+'''' );

        FETCH NEXT FROM Codigo INTO @code, @Region

    END

CLOSE Codigo
DEALLOCATE Codigo

-- *****

--EXECUTE('CREATE TABLE n2kmanager.dbo.Sites_Change_Criteria(
--        [SiteCode] [nvarchar](9) NULL,
--        '+@Release+' int NULL,
--        '+@Release1+' int NULL
--        )');

EXECUTE('insert            into            n2kmanager.dbo.Sites_Change_Criteria
(SiteCode, '+@Release+', '+@Release1+')

                                SELECT R1.[SCICode]
                                ,R2.Criteria
                                ,R1.Criteria
FROM n2kmanager.[dbo].'+ @ReleaseName1 +' as
R1
                                ,n2kmanager.[dbo].'+ @ReleaseName2 +' as R2
                                WHERE
                                R1.SCICode = R2.SCICode
                                and R1.Criteria != R2.Criteria');

-- *****

IF object_id('n2kmanager.dbo.Sites_Change_Criteria_Temporal' ) IS NOT NULL
        EXECUTE('DROP TABLE n2kmanager.dbo.Sites_Change_Criteria_Temporal');

EXECUTE('CREATE TABLE n2kmanager.dbo.Sites_Change_Criteria_Temporal(
                                [SiteCode] [nvarchar](9) NULL )');

EXECUTE('insert into n2kmanager.dbo.Sites_Change_Criteria_Temporal (SiteCode)
        Select SiteCode
        FROM n2kmanager.dbo.Sites_Change_Criteria_' + @site + '
        ');

if ((select count(*) from n2kmanager.dbo.Sites_Change_Criteria_Temporal) = 0 )
        exec('DROP TABLE n2kmanager.dbo.Sites_Change_Criteria_' + @site);

```

```
-- *****

                                FETCH NEXT FROM SitesChange_Criteria INTO @site

                                END

                                CLOSE SitesChange_Criteria
                                DEALLOCATE SitesChange_Criteria

--*****

IF object_id('n2kmanager.dbo.Change_Criteria1' ) IS NOT NULL
    EXECUTE('DROP TABLE n2kmanager.dbo.Change_Criteria1');

CREATE TABLE n2kmanager.[dbo].[Change_Criteria1](
    [SiteCode] [nvarchar](9) NULL,
    [New_Criteria] [int] NULL
)
insert into n2kmanager.dbo.Change_Criteria1(SiteCode,New_Criteria)
SELECT [SiteCode]
      ,[END2010]
FROM [n2kmanager].[dbo].[Sites_Change_Criteria]
where End2010='1'

IF object_id('n2kmanager.dbo.Change_Criteria2' ) IS NOT NULL
    EXECUTE('DROP TABLE n2kmanager.dbo.Change_Criteria2');

CREATE TABLE n2kmanager.[dbo].[Change_Criteria2](
    [SiteCode] [nvarchar](9) NULL,
    [New_Criteria] [int] NULL
)
insert into n2kmanager.dbo.Change_Criteria2(SiteCode,New_Criteria)
SELECT [SiteCode]
      ,[END2010]
FROM [n2kmanager].[dbo].[Sites_Change_Criteria]
where End2010='2'

IF object_id('n2kmanager.dbo.Change_Criteria3' ) IS NOT NULL
    EXECUTE('DROP TABLE n2kmanager.dbo.Change_Criteria3');

CREATE TABLE n2kmanager.[dbo].[Change_Criteria3](
    [SiteCode] [nvarchar](9) NULL,
    [New_Criteria] [int] NULL
)
insert into n2kmanager.dbo.Change_Criteria3(SiteCode,New_Criteria)
SELECT [SiteCode]
      ,[END2010]
FROM [n2kmanager].[dbo].[Sites_Change_Criteria]
where End2010='1'
```

```

IF object_id('n2kmanager.dbo.Change_Criteria4') IS NOT NULL
    EXECUTE('DROP TABLE n2kmanager.dbo.Change_Criteria4');

CREATE TABLE n2kmanager.[dbo].[Change_Criteria4](
    [SiteCode] [nvarchar](9) NULL,
    [New_Criteria] [int] NULL
)
insert into n2kmanager.dbo.Change_Criteria4(SiteCode,New_Criteria)
SELECT [SiteCode]
,[END2010]
FROM [n2kmanager].[dbo].[Sites_Change_Criteria]
where End2010='4'

IF object_id('n2kmanager.dbo.Change_Criteria5' ) IS NOT NULL
    EXECUTE('DROP TABLE n2kmanager.dbo.Change_Criteria5');

CREATE TABLE n2kmanager.[dbo].[Change_Criteria5](
    [SiteCode] [nvarchar](9) NULL,
    [New_Criteria] [int] NULL
)
insert into n2kmanager.dbo.Change_Criteria5(SiteCode,New_Criteria)
SELECT [SiteCode]
,[END2010]
FROM [n2kmanager].[dbo].[Sites_Change_Criteria]
where End2010='5'

IF object_id('n2kmanager.dbo.Change_Criteria6' ) IS NOT NULL
    EXECUTE('DROP TABLE n2kmanager.dbo.Change_Criteria6');

CREATE TABLE n2kmanager.[dbo].[Change_Criteria6](
    [SiteCode] [nvarchar](9) NULL,
    [New_Criteria] [int] NULL
)
insert into n2kmanager.dbo.Change_Criteria6(SiteCode,New_Criteria)
SELECT [SiteCode]
,[END2010]
FROM [n2kmanager].[dbo].[Sites_Change_Criteria]
where End2010='6'

-- *****
end

```

3. State the reason for the change of criteria. The last step is to give a definition for the changes on criteria. This is convention in use for the change of criteria for values:

✓ 1 to 2 'Priority feature deleted'

- ✓ 1 to 3 'Priority feature deleted'
- ✓ 1 to 4 'Priority feature deleted'
- ✓ 1 to 5 'Priority feature deleted'

- ✓ 2 to 1 'Priority feature added'
- ✓ 2 to 3 'Only record of a feature in MS deleted'
- ✓ 2 to 4 'Only record of a feature in MS deleted'
- ✓ 2 to 5 'Only record of a feature in MS deleted'
- ✓ 3 to 1 'Priority feature added'
- ✓ 3 to 2 'Only record of a feature in MS added'
- ✓ 3 to 4 'High quality feature deleted'
- ✓ 3 to 5 'High quality feature deleted'
- ✓ 4 to 1 'Priority feature added'
- ✓ 4 to 2 'Only record of a feature in MS added'
- ✓ 4 to 3 'High quality feature added'
- ✓ 4 to 5 'Feature(s) deleted high diversity criterion not applicable'
- ✓ 5 to 1 'Priority feature added'
- ✓ 5 to 2 'Only record of a feature in MS added'
- ✓ 5 to 3 'High quality feature added'
- ✓ 5 to 4 'Feature(s) added high diversity criterion applicable'

- ✓ 1 to 6 'All significant (at least C) features have been removed'
- ✓ 2 to 6 'All significant (at least C) features have been removed'
- ✓ 3 to 6 'All significant (at least C) features have been removed'
- ✓ 4 to 6 'All significant (at least C) features have been removed'
- ✓ 5 to 6 'All significant (at least C) features have been removed'

- ✓ 6 to 1 'Priority feature added'
- ✓ 6 to 2 'Only record of a feature in MS added'
- ✓ 6 to 3 'High quality feature added'
- ✓ 6 to 4 'Feature(s) added high diversity criterion applicable'

✓ 6 to 5 'Features (at least C have been added'