付録 1　モデルの実装

コンピュータのなかにおいて仮想国家のモデルは、(株)構造計画研究所開発・製作の汎用型マルチエージェント・シミュレータartisocを使って実装される。以下に示すのは、この実装に用いるファイルの内容をテキスト形式で書き出したものである。モデル本体の構造やルールだけではなく、インタフェイスの定義や設定、外部ファイルからの入力データの読み込み、出力データの構成やその表示の設定など、シミュレーションの設定・実行・出力に関わるあらゆる情報が含まれている。

なお、artisocは、エージェントを定義しその行動を記述するための独自の仕様のクラスを備えているが、計算速度が遅くなるなどの技術的な理由で、ここではこの仕様をほとんど活用していない。その代わり、PopCellエージェントとRulerエージェントの変数は配列によって個別に管理し、これら変数の変化を定める両エージェントの行動ルールは、”Universe”というartisocが定める特殊なエージェントを使って一括して実行している。もちろん、このような設計上の選択が、本書第二章で記述した仮想国家のモデルの内実に実質的な変更をもたらすことはない。

```
//=================================================================/
//                                                               /
// artisoc                                        /
//                                               /
//=================================================================/


//---------------------------------------------------
// Component Tree
//---------------------------------------------------
UNIVERSE {
    Space LatticeSpace(0,0),Square_2D(50, 50, !Loop, North, 1){
        Dim Territory(0, 0)[50][50][1] As Integer = (2500*0);
        Dim Area(0, 0)[50][50][1] As Double = (2500*0.0);
        Dim Pop(0, 0)[50][50][1] As Double = (2500*0.0);
        Dim Traits(0, 0)[50][50][1] As String = (2500*"");
        Dim Comp(0, 0)[50][50][1] As String = (2500*"");
        Dim Region(0, 0)[50][50][1] As Integer = (2500*0);
```

```
        Dim Ruler(0, 0)[50][50][1] As Integer = (2500*0);
        Dim Deployment(0, 0)[50][50][1] As String = (2500*"");
        Dim Frontline(0, 0)[50][50][1] As Integer = (2500*0);
        Dim CDistance(0, 0)[50][50][1] As String = (2500*"");
        Dim Color1(0, 0)[50][50][1] As Double = (2500*0.0);
        Dim Color2(0, 0)[50][50][1] As Double = (2500*0.0);
        AgtType Display(0, 0)[0] {
            Dim ID(0, 0) As Integer = (0);
            Dim X(0, 0) As Double = (0.0);
            Dim Y(0, 0) As Double = (0.0);
            Dim Layer(0, 0) As Integer = (0);
            Dim Direction(0, 0) As Double = (0.0);
            Dim Color(0, 0) As Integer = (0);
        }
    }
    Dim Country(0, 0) As String = ("");
    Dim Control1(0, 0)[14] As Integer = (14*0);
    Dim Control2(0, 0)[5] As Integer = (5*0);
    Dim Capital(0, 0)[1000][2] As Integer = (2000*0);
    Dim RPC(0, 0) As Double = (0.0);
    Dim SpecifiedInsgt(0, 0) As String = ("");
    Dim NumOfRulers(0, 0) As Integer = (0);
    Dim Relationship(0, 0) As Integer = (0);
    Dim TraitsOfRulers(0, 0)[1000] As String = (1000*"");
    Dim MobilizationFactors(0, 0)[1000][2] As Double = (2000*0.0);
    Dim ResourcesOfRulers(0, 0)[1000] As Double = (1000*0.0);
    Dim WeightNormalizers(0, 0)[1000] As Double = (1000*0.0);
    Dim RulerControl(0, 0)[7] As Double = (7*0.0);
    Dim PotentialControl(0, 0)[4] As Double = (4*0.0);
    Dim RelationalControl(0, 0)[2] As Double = (2*0.0);
    Dim Noise(0, 0) As Double = (0.0);
    Dim TimeManager(0, 0) As Integer = (0);
    Dim BasicInfo(0, 0)[6] As Double = (6*0.0);
    Dim Counter(0, 0)[4] As Integer = (4*0);
    Dim OrderIndex(0, 0)[2] As Double = (2*0.0);
    Dim Govt(0, 0) As Integer = (0);
    Dim GrossPotential(0, 0) As Double = (0.0);
    Dim LatentRulers(0, 0)[1000] As Boolean = (1000*0);
    Dim CellDiv(0, 0)[1000] As Integer = (1000*0);
    Dim TerDiv(0, 0)[1000] As Double = (1000*0.0);
    Dim PopDiv(0, 0)[1000] As Double = (1000*0.0);
    Dim SubPopDiv(0, 0)[20] As Double = (20*0.0);
    Dim RulerSet(0, 0)[1000] As String = (1000*"");
    Dim RulerMonitor(0, 0)[2][4] As Double = (8*0.0);
    Dim RulerMonitor_Gr(0, 0)[2] As String = (2*"");
    Dim MajRulers(0, 0)[10] As String = (10*"");
    Dim MajRulers_TerDiv(0, 0)[10] As Double = (10*0.0);
    Dim MajRulers_Traits(0, 0)[10] As String = (10*"");
    Dim MajRulers_MblLv(0, 0)[10] As Double = (10*0.0);
    Dim MajRulers_Resources(0, 0)[10] As Double = (10*0.0);
    Dim MajRulers_Group(0, 0)[10] As String = (10*"");
    Dim FilesOut(0, 0) As Boolean = (0);
    Dim Snapshot(0, 0) As Boolean = (0);
}


//-----------------------------------------------------
// Agent Variable Initial Data
//-----------------------------------------------------
Initial_Value {
    UNIVERSE.COUNTRY = 1("ETH");
```

```
    UNIVERSE.CONTROL1 = 1(14*0);
    UNIVERSE.CONTROL2 = 1(0, 3*2, 0);
    UNIVERSE.RPC = 1(0.0);
    UNIVERSE.SPECIFIEDINSGT = 1("");
    UNIVERSE.NUMOFRULERS = 1(100);
    UNIVERSE.RELATIONSHIP = 1(0);
    UNIVERSE.TRAITSOFRULERS = 1(1000*"");
    UNIVERSE.RULERCONTROL = 1(1.0, 0.0, 1.0, 0.0, 2.0, 1.0, 0.0);
    UNIVERSE.POTENTIALCONTROL = 1(0.2, 0.0, 2*100.0);
    UNIVERSE.RELATIONALCONTROL = 1(1.0, 0.0);
    UNIVERSE.NOISE = 1(0.2);
    UNIVERSE.TIMEMANAGER = 1(500);
    UNIVERSE.SUBPOPDIV = 1(20*0.0);
    UNIVERSE.FILESOUT = 1(1);
    UNIVERSE.SNAPSHOT = 1(0);
    UNIVERSE.LATTICESPACE.FRONTLINE = 1(2500*0);
}


//-------------------------------------------------------
// Agent Rule
//-------------------------------------------------------
#begin_rule UNIVERSE
Univ_Init{

        Dim i As Integer
        Dim j As Integer
        Dim k As Integer
        Dim l As Integer
        Dim m As Integer
        Dim n As Integer
        Dim l_x As Integer
        Dim l_y As Integer
        Dim r As Double
        Dim str1 As String
        Dim str2 As String
        Dim str3 As String
        Dim list( 5 ) As String
        Dim data( 20 ) As String
        Dim data2 As String

        // コンソール出力初期化
        ClearConsoleScreen( )
        str1 = "Welcome to " & Universe.Country & "!!"
        PrintLn( str1 )
        PrintLn( "Click on the Map if you want some more information." )

        // 仮想国家の読み込みと構成
        str1 = Universe.Country & "//"

        // ヘッダーファイルの読み込み
        str2 = str1 & "hdr.txt"
        OpenFile( str2, 1, 1 )
        For i = 0 To 2
                Universe.Control1( i ) = CInt( ReadFile( 1 ) )
        Next i
        For i = 0 To Universe.Control1( 1 ) - 1
                Universe.Control1( 3 + i ) = CInt( ReadFile( 1 ) )
        Next i
        For i = 0 To Universe.Control1( 1 ) - 1
                Universe.Control1( 6 + i ) = CInt( ReadFile( 1 ) )
```

```
        Next i
        For i = 0 To 3
                Universe.Control1( 9 + i ) = CInt( ReadFile( 1 ) )
        Next i
        For i = 0 To 1
                Universe.Capital( 0, i ) = Universe.Control1( 9 + i ) + CInt( ReadFile( 1 ) )
        Next i

        // シミュレーションで属性データをフルに使用する場合の設定
        If Universe.Control2( 0 ) == 1 Then
                For i = 0 To Universe.Control1( 1 ) - 1
                        Universe.Control2( 1 + i ) = Universe.Control1( 2 )
                Next i
        End If

        // 初期政府の属性読み込み
        k = 0
        For i = 0 To Universe.Control1( 1 ) - 1
                k = k + Universe.Control2( 1 + i ) + 1
        Next i
        If Len( Universe.TraitsOfRulers( 0 ) ) <> k Then
                Universe.TraitsOfRulers( 0 ) = ""
                For i = 0 To Universe.Control1( 1 ) - 1
                        str3 = ReadFile( 1 )
                        If Len( str3 ) < Universe.Control1( 2 ) Then
                                str3 = "0" & str3
                        End If
                        Universe.TraitsOfRulers( 0 ) = Universe.TraitsOfRulers( 0 ) & Left( str3,
Universe.Control2( 1 + i ) ) & ","
                Next i
        Else
                For i = 0 To Universe.Control1( 1 ) - 1
                        ReadFile( 1 )
                Next i
        End If

        // 一人当たり資源量の読み込み
        If Universe.RPC <= 0 Then
                Universe.RPC = CDbl( ReadFile( 1 ) )
        End If

        CloseFile( 1 )

        // 基本情報の初期化
        For i = 0 To 5
                Universe.BasicInfo( i ) = 0
                list( i ) = ""
        Next i

        // 空間データのファイルをオープン
        str2 = str1 & "area.txt"
        OpenFile( str2, 1, 1 )
        str2 = str1 & "pop.txt"
        OpenFile( str2, 2, 1 )
        m = 3
        For i = 0 To Universe.Control1( 1 ) - 1
                For j = 0 To Universe.Control1( 6 + i ) - 1
                        str2 = str1 & "trait" & CStr( 1 + i ) & "_" & CStr( 1 + j ) & ".txt"
                        OpenFile( str2, m, 1 )
                        str2 = str1 & "pro" & CStr( 1 + i ) & "_" & CStr( 1 + j ) & ".txt"
                        OpenFile( str2, m + 1, 1 )
```

```
                        m = m + 2
                Next j
        Next i

        // 「分権化」指定の場合地域コードデータのファイルもオープン
        If Universe.Relationship == 2 Then
                str2 = str1 & "regions.txt"
                OpenFile( str2, 21, 1 )
        End If

        // 空間データの読み込みと集計
        For i = 0 To Universe.Control1( 11 )
                For j = 0 To Universe.Control1( 12 )
                        For k = 0 To Universe.Control1( 0 ) - 1
                                data( k ) = ReadFile( 1 + k )
                        Next k

                        // 「分権化」指定の場合地域コードデータも読み込み
                        If Universe.Relationship == 2 Then
                                data2 = ReadFile( 21 )
                        End If

                        // 住民が存在する格子点にデータを入力
                        If CDbl( data( 1 ) ) > 0 Then
                                l_x = Universe.Control1( 9 ) + i
                                l_y = Universe.Control1( 10 ) + j
                                Universe.LatticeSpace.Territory( l_x, l_y, 0 ) = 1
                                Universe.LatticeSpace.Area( l_x, l_y, 0 ) = CDbl( data( 0 ) )
                                Universe.LatticeSpace.Pop( l_x, l_y, 0 ) = CDbl( data( 1 ) )

                                // 属性データの入力
                                Universe.LatticeSpace.Traits( l_x, l_y, 0 ) = ""
                                Universe.LatticeSpace.Comp( l_x, l_y, 0 ) = ""
                                m = 2
                                For k = 0 To Universe.Control1( 1 ) - 1
                                        For l = 0 To Universe.Control1( 6 + k ) - 1
                                                If Len( data( m ) ) < Universe.Control1( 2 ) AND data(
m ) <> "0" Then
                                                        data( m ) = "0" & data( m )
                                                End If
                                                Universe.LatticeSpace.Traits( l_x, l_y, 0 ) =
Universe.LatticeSpace.Traits( l_x, l_y, 0 ) & Left( data( m ), Universe.Control2( 1 + k ) ) & ","
                                                Universe.LatticeSpace.Comp( l_x, l_y, 0 ) =
Universe.LatticeSpace.Comp( l_x, l_y, 0 ) & data( m + 1 ) & ","

                                                // 属性の類型数のカウント
                                                If InStr( 1, list( k ), Left( data( m ), Universe.Control2(
1 + k ) ) ) == 0 Then
                                                        list( k ) = list( k ) & Left( data( m ),
Universe.Control2( 1 + k ) ) & ","
                                                        Universe.BasicInfo( 3 + k ) =
Universe.BasicInfo( 3 + k ) + 1
                                                End If

                                                m = m + 2
                                        Next l
                                Next k

                                // 基本情報の集計
                                Universe.BasicInfo( 0 ) = Universe.BasicInfo( 0 ) + 1
```

```
                                        Universe.BasicInfo( 1 ) = Universe.BasicInfo( 1 ) +
Universe.LatticeSpace.Area( l_x, l_y, 0 )
                                        Universe.BasicInfo( 2 ) = Universe.BasicInfo( 2 ) +
Universe.LatticeSpace.Pop( l_x, l_y, 0 )

                                        // 「分権化」指定の場合地域コードの入力と地域別人口の集計
                                        If Universe.Relationship == 2 Then
                                                n = CInt( data2 )
                                                Universe.LatticeSpace.Region( l_x, l_y, 0 ) = n
                                                Universe.SubPopDiv( n ) = Universe.SubPopDiv( n ) +
Universe.LatticeSpace.Pop( l_x, l_y, 0 )
                                        End If

                                End If

                        Next j
                Next i

                // 空間データのファイルをクローズ
                For i = 0 To Universe.Control1( 0 ) - 1
                        CloseFile( 1 + i )
                Next i

                // 「分権化」指定の場合地域コードデータのファイルもクローズ
                If Universe.Relationship == 2 Then
                        CloseFile( 21 )
                End If

                // 初期政府に関わる他の変数の初期化
                Universe.MobilizationFactors( 0, 0 ) = Universe.RulerControl( 0 )
                Universe.MobilizationFactors( 0, 1 ) = Universe.RulerControl( 1 )
                Universe.CellDiv( 0 ) = Universe.BasicInfo( 0 )
                Universe.TerDiv( 0 ) = 1
                Universe.PopDiv( 0 ) = Universe.BasicInfo( 2 )
                If Universe.Relationship <>2 Then
                        Universe.ResourcesOfRulers( 0 ) = Universe.MobilizationFactors( 0, 0 ) * ( ( Universe.RPC *
Universe.PopDiv( 0 ) ) / 20000000 ) + Universe.MobilizationFactors( 0, 1 )
                Else
                        Universe.ResourcesOfRulers( 0 ) = Universe.MobilizationFactors( 0, 0 ) * ( ( Universe.RPC *
Universe.SubPopDiv( 0 ) ) / 20000000 ) + Universe.MobilizationFactors( 0, 1 )
                End If
                Universe.RulerSet( 0 ) = ""

                // 「分権化」指定の場合下位Rulerのデータのファイルをオープン
                If Universe.Relationship == 2 Then
                        str2 = str1 & "units.txt"
                        OpenFile( str2, 2, 1 )

                        // 下位Ruler数を読み込み指定Ruler数と矛盾きたすならエラー
                        Universe.Control1( 13 ) = CInt( ReadFile( 2 ) )
                        If Universe.Control1( 13 ) + 1 >= Universe.NumOfRulers Then
                                ExitSimulationMsg( "The Number of Rulers Specified is Insufficient." )
                        End If

                End If

                // 反政府Rulerの属性・動員パラメータ等の初期化
                For i = 0 To Universe.NumOfRulers - 2

                        // 指定がある場合反政府Rulerのカスタマイズ
                        If 1 + i == Universe.NumOfRulers - 1 AND Universe.SpecifiedInsgt <> "" Then
```

```
                        // 指定ファイルをオープン
                        str2 = str1 & Universe.SpecifiedInsgt & ".txt"
                        OpenFile( str2, 1, 1 )

                        // 首都情報の読み込み
                        Universe.Capital( 1 + i, 0 ) = Universe.Control1( 9 ) + CInt( ReadFile( 1 ) )
                        Universe.Capital( 1 + i, 1 ) = Universe.Control1( 10 ) + CInt( ReadFile( 1 ) )

                        // 属性の読み込み
                        For j = 0 To Universe.Control1( 1 ) - 1
                                str3 = ReadFile( 1 )
                                If Len( str3 ) < Universe.Control1( 2 ) Then
                                        str3 = "0" & str3
                                End If
                                Universe.TraitsOfRulers( 1 + i ) = Universe.TraitsOfRulers( 1 + i ) &
Left( str3, Universe.Control2( 1 + j ) ) & ","
                        Next j

                        // ファイルをクローズ
                        CloseFile( 1 )

                        // 動員パラメータの初期化
                        Universe.MobilizationFactors( 1 + i, 0 ) = Universe.RulerControl( 5 )
                        Universe.MobilizationFactors( 1 + i, 1 ) = Universe.RulerControl( 6 )

                        // 保有資源の初期化
                        Universe.ResourcesOfRulers( 1 + i ) = Universe.MobilizationFactors( 1 + i, 1 )

                        // 所属集合の初期化
                        Universe.RulerSet( 1 + i ) = ""

                // 「分権化」指定の場合指定数まで下位Rulerの生成
                ElseIf Universe.Relationship == 2 AND 1 + i <= Universe.Control1( 13 ) Then

                        // 識別コードの読み込み
                        Universe.RulerSet( 1 + i ) = ReadFile( 2 )

                        // 首都情報の読み込み
                        Universe.Capital( 1 + i, 0 ) = Universe.Control1( 9 ) + CInt( ReadFile( 2 ) )
                        Universe.Capital( 1 + i, 1 ) = Universe.Control1( 10 ) + CInt( ReadFile( 2 ) )

                        // 属性の読み込み
                        For j = 0 To Universe.Control1( 1 ) - 1
                                str3 = ReadFile( 2 )
                                If Len( str3 ) < Universe.Control1( 2 ) Then
                                        str3 = "0" & str3
                                End If
                                Universe.TraitsOfRulers( 1 + i ) = Universe.TraitsOfRulers( 1 + i ) &
Left( str3, Universe.Control2( 1 + j ) ) & ","
                        Next j

                        // 動員パラメータの初期化
                        Universe.MobilizationFactors( 1 + i, 0 ) = Universe.MobilizationFactors( 0, 0 )
                        Universe.MobilizationFactors( 1 + i, 1 ) = 0

                        // 保有資源の初期化と上位Rulerへの資源加算
                        Universe.ResourcesOfRulers( 1 + i ) = Universe.RelationalControl( 1 ) *
Universe.MobilizationFactors( 0, 0 ) * ( ( Universe.RPC * Universe.SubPopDiv( 1 + i ) ) / 20000000 )
```

```
                        Universe.ResourcesOfRulers( 0 ) = Universe.ResourcesOfRulers( 0 ) + ( 1 -
Universe.RelationalControl( 1 ) ) * Universe.MobilizationFactors( 0, 0 ) * ( ( Universe.RPC *
Universe.SubPopDiv( 1 + i ) ) / 20000000 )

                    // それ以外はランダム生成
                    Else

                        // 属性のランダム生成
                        Universe.TraitsOfRulers( 1 + i ) = ""
                        Do While 1
                            l_x = Universe.Control1( 9 ) + Round( Universe.Control1( 11 ) * Rnd( ) )
                            l_y = Universe.Control1( 10 ) + Round( Universe.Control1( 12 ) * Rnd( )
)
                            If Universe.LatticeSpace.Territory( l_x, l_y, 0 ) == 1 Then
                                break
                            End If
                        Loop
                        m = 0
                        For j = 0 To Universe.Control1( 1 ) - 1
                            r = Rnd( )
                            For k = 0 To Universe.Control1( 6 + j ) - 1
                                r = r - CDbl( GetToken( Universe.LatticeSpace.Comp( l_x, l_y,
0 ), m + k ) )
                                If r < 0 Then
                                    str3 = GetToken( Universe.LatticeSpace.Traits( l_x,
l_y, 0 ), m + k )
                                    break
                                End If
                            Next k
                            m = m +  Universe.Control1( 6 + j )
                            n = Universe.Control2( 1 + j ) / 2
                            For k = 0 To n - 1
                                If Mid( str3, 2 * k + 1, 2 ) == "00" Then
                                    For l = k To n - 1
                                        Universe.TraitsOfRulers( 1 + i ) =
Universe.TraitsOfRulers( 1 + i ) & "00"
                                    Next l
                                    break
                                ElseIf Rnd( ) < 0.5 Then
                                    For l = k To n - 1
                                        Universe.TraitsOfRulers( 1 + i ) =
Universe.TraitsOfRulers( 1 + i ) & "**"
                                    Next l
                                    break
                                Else
                                    Universe.TraitsOfRulers( 1 + i ) =
Universe.TraitsOfRulers( 1 + i ) & Mid( str3, 2 * k + 1, 2 )
                                End If
                            Next k
                            Universe.TraitsOfRulers( 1 + i ) = Universe.TraitsOfRulers( 1 + i ) & ","
                        Next j

                        // 反政府Rulerの「首都」の指定
                        Universe.Capital( 1 + i, 0 ) = l_x
                        Universe.Capital( 1 + i, 1 ) = l_y

                        // 動員パラメータの初期化
                        Universe.MobilizationFactors( 1 + i, 0 ) = 1 + ( Universe.RulerControl( 2 ) - 1 ) *
Rnd( )
                        Universe.MobilizationFactors( 1 + i, 1 ) = Universe.RulerControl( 3 )
```

```
                    // 保有資源の初期化
                    Universe.ResourcesOfRulers( 1 + i ) = Universe.MobilizationFactors( 1 + i, 1 )

                    // 所属集合の初期化
                    Universe.RulerSet( 1 + i ) = ""

                End If

                // 潜在的Rulerの指定
                Universe.LatentRulers( 1 + i ) = True

        Next i

        // 「分権化」指定の場合下位Rulerのデータのファイルをクローズ
        If Universe.Relationship == 2 Then
                CloseFile( 2 )
        End If

        // 初期政府の資源配備、属性間の距離や表示色など他の空間データの初期化
        For i = 0 To GetWidthSpace( Universe.LatticeSpace ) - 1
                For j = 0 To GetHeightSpace( Universe.LatticeSpace ) - 1
                        If Universe.LatticeSpace.Territory( i, j, 0 ) == 0 Then
                                Universe.LatticeSpace.Ruler( i, j, 0 ) = -1
                                If Universe.Relationship == 2 Then
                                        Universe.LatticeSpace.Region( i, j, 0 ) = -1
                                End If
                        Else

                                // 初期政府の資源配備とウェイト標準化因子の計算
                                Universe.LatticeSpace.Deployment( i, j, 0 ) = "0," & CStr( Log(
Universe.LatticeSpace.Pop( i, j, 0 ) ) * Universe.ResourcesOfRulers( 0 ) )
                                Universe.WeightNormalizers( 0 ) = Universe.WeightNormalizers( 0 ) +
Log( Universe.LatticeSpace.Pop( i, j, 0 ) )

                                // 「分権化」指定の場合下位Rul-
erによる資源配備の追加とウェイト標準化因子の計算
                                If Universe.Relationship == 2 Then
                                        k = Universe.LatticeSpace.Region( i, j, 0 )
                                        If k > 0 Then
                                                Universe.LatticeSpace.Deployment( i, j, 0 ) =
Universe.LatticeSpace.Deployment( i, j, 0 ) & "," & CStr( Log( Universe.LatticeSpace.Pop( i, j, 0 ) ) *
Universe.ResourcesOfRulers( k ) )
                                                Universe.WeightNormalizers( k ) =
Universe.WeightNormalizers( k ) + Log( Universe.LatticeSpace.Pop( i, j, 0 ) )
                                        End If
                                End If

                                // Rulerの属性との距離の計算と記録
                                Universe.LatticeSpace.CDistance( i, j, 0 ) = ""
                                For k = 0 To Universe.NumOfRulers - 1
                                        Universe.LatticeSpace.CDistance( i, j, 0 ) =
Universe.LatticeSpace.CDistance( i, j, 0 ) & CStr( StdCedermanDistance( Universe.LatticeSpace.Traits( i, j, 0 ),
Universe.TraitsOfRulers( k ), Universe.LatticeSpace.Comp( i, j, 0 ) ) ) & ","
                                Next k

                                // 属性色の設定
                                l = 0
                                m = Universe.Control2( 4 )
                                If m > 0 Then
                                        For k = 0 To m - 1
                                                l = l + Universe.Control1( 6 + k )
```

```
                                        Next k
                                    End If
                                    str3 = GetToken( Universe.LatticeSpace.Traits( i, j, 0 ), l )
                                    Universe.LatticeSpace.Color1( i, j, 0 ) = 0.2 + 0.8 * ( CDbl(
Universe.Control1( 3 + m ) ) - CDbl( str3 ) * 10 ^ ( Universe.Control1( 2 ) - Universe.Control2( 1 + m ) ) ) /
CDbl( Universe.Control1( 3 + m ) )

                                    // 状態色＝Rulerの色の初期化
                                    Universe.LatticeSpace.Color2( i, j, 0 ) = 0.85
                                    If i == Universe.Capital( 0, 0 ) AND j == Universe.Capital( 0, 1 ) Then
                                        Universe.LatticeSpace.Color2( i, j, 0 ) = 1
                                    End If

                                End If

                            Next j
                        Next i

                        // カウンターの初期化
                        Universe.Counter( 1 ) = 1

                        // ファイル出力の準備とパラメータ環境等の書き出し
                        If Universe.FilesOut == True Then
                            str1 = "Output//" & Universe.Country & "//setting.txt"
                            str3 = Universe.SpecifiedInsgt
                            If str3 == "" Then
                                str3 = "void"
                            End If
                            str2 = TimeToStr( GetRealTime( ), "yyyyMMddHHmm " ) & Universe.Country & " " & CStr(
Universe.RPC ) & " " & Universe.TraitsOfRulers( 0 ) & " " & str3 & " " & CStr( Universe.NumOfRulers ) & " " &
CStr( Universe.Relationship ) & " " & CStr( Universe.RulerControl( 0 ) ) & " " & CStr( Universe.RulerControl( 1
) ) & " " & CStr( Universe.RulerControl( 5 ) ) & " " & CStr( Universe.RulerControl( 6 ) ) & " " & CStr(
Universe.RulerControl( 2 ) ) & " " & CStr( Universe.RulerControl( 4 ) ) & " " & CStr(
Universe.RelationalControl( 0 ) ) & " " & CStr( Universe.RelationalControl( 1 ) ) & " " & CStr(
Universe.PotentialControl( 0 ) ) & " " & CStr( Universe.PotentialControl( 1 ) ) & " " & CStr( Universe.Noise ) & "
"
                            OpenFile( str1, 1, 3 )
                            WriteLnFile( 1, "Time Country PerCapitaResources InitGovtTraits SInsgt NofRulers Rela-
tionship InitGovtMobilization ExInputToInitGovt SInsgtMobilization ExInputToSInsgt MobilizationLimit
WeightToFront ThreatThreshold Decentralization CoercionEffect MobilizationEffect Noise " )
                            WriteLnFile( 1, str2 )
                            CloseFile( 1 )
                            str1 = "Output//" & Universe.Country & "//rulers.txt"
                            OpenFile( str1, 1, 3 )
                            WriteLnFile( 1, "ID Group Capital Traits MobilizationLv ExInput " )
                            For i = 0 To Universe.NumOfRulers - 1
                                str3 = Universe.RulerSet( i )
                                If Universe.Relationship == 2 AND i == 0 Then
                                    str3 = "f"
                                ElseIf str3 == "" Then
                                    str3 = "void"
                                End If
                                str2 = CStr( i ) & " " & str3 & " " & CStr( Universe.Capital( i, 0 ) -
Universe.Control1( 9 ) ) & "," & CStr( Universe.Capital( i, 1 ) - Universe.Control1( 10 ) ) & " " &
Universe.TraitsOfRulers( i ) & " " & CStr( Universe.MobilizationFactors( i, 0 ) ) & " " & CStr(
Universe.MobilizationFactors( i, 1 ) ) & " "
                                WriteLnFile( 1, str2 )
                            Next i
                            CloseFile( 1 )
                            str1 = "Output//" & Universe.Country & "//dynamics.txt"
```

```
                str2 =  "Simulation Period Entropy ENofRulers NofPresentRulers NofEmergentRulers Init-
GovtPresent SInsgtPresent InitGovtCells InitGovtTer InitGovtPop InitGovtResources InitGovtGroup SInsgtCells
SInsgtTer SInsgtPop SInsgtResources SInsgtGroup CurGovt GovtTraits GovtMobilization GovtTerritory
GovtResources GovtGroup "
                For i = 0 To 8
                        str3 = "Insgt" & CStr( 1 + i )
                        str2 = str2 & str3 & " " & str3 & "Traits " & str3 & "Mobilization " & str3 & "Terri-
tory " & str3 & "Resources " & str3 & "Group "
                Next i
                str2 =str2 & "GrossPotential "
                OpenFile( str1, 1, 3 )
                WriteLnFile( 1, str2 )
                CloseFile( 1 )
        End If

}

Univ_Step_Begin{

        Dim i As Integer
        Dim j As Integer
        Dim t_x As Integer
        Dim t_y As Integer
        Dim str As String

        // 指定の場合初期画面の撮影
        If Universe.Snapshot == True AND GetCountStep( ) == 1 Then
                str = "Output//" & Universe.Country & "//Pics//" & TimeToStr( GetRealTime( ),
"yyyyMMddHHmmss_" ) & CStr( GetCountSimulationNumber( ) ) & "_" & CStr( GetCountStep( ) ) & ".jpg"
                ScreenShot( str )
        End If

        // 総ポテンシャルの初期化
        Universe.GrossPotential = 0

        // 住民が存在する格子点で状態更新
        For i = 0 To Universe.Control1( 11 )
                t_x = Universe.Control1( 9 ) + i
                For j = 0 To Universe.Control1( 12 )
                        t_y = Universe.Control1( 10 ) + j
                        If Universe.LatticeSpace.Territory( t_x, t_y, 0 ) == 1 Then
                                GibbsSampler( t_x, t_y )
                        End If
                Next j
        Next i

}

Univ_Step_End{

        Dim i As Integer
        Dim j As Integer
        Dim k As Integer
        Dim l As Integer
        Dim m As Integer
        Dim n As Integer
        Dim l_x As Integer
        Dim l_y As Integer
        Dim n_x As Integer
        Dim n_y As Integer
        Dim e As Double
```

```
	Dim r As Double
	Dim s As Double
	Dim bk( 1000 ) As Double
	Dim one As Agt
	Dim str1 As String
	Dim str2 As String
	Dim list( 1000 ) As String
	Dim t As Boolean

	// 変数初期化
	For i = 0 To 9
		Universe.MajRulers( i ) = ""
		Universe.MajRulers_TerDiv( i ) = 0
		Universe.MajRulers_Traits( i ) = ""
		Universe.MajRulers_MblLv( i ) = 0
		Universe.MajRulers_Resources( i ) = 0
		Universe.MajRulers_Group( i ) = ""
	Next i
	Universe.Counter( 0 ) = 0

	// Rulerごとの処理――資源の動員、および秩序指標や主要Rulerの出力指標の計算
	r = 0
	For i = 0 To Universe.NumOfRulers - 1

		// Rulerの動員パラメータの更新と資源の動員
		bk( i ) = Universe.ResourcesOfRulers( i )
		If i == 0 Then
			Universe.MobilizationFactors( i, 1 ) = 0
			If Universe.Govt == 0 Then
				Universe.MobilizationFactors( i, 1 ) = Universe.RulerControl( 1 )
			End If
		ElseIf i == Universe.NumOfRulers - 1 AND Universe.SpecifiedInsgt <> "" Then
			Universe.MobilizationFactors( i, 1 ) = Universe.RulerControl( 6 )
		End If
		If Universe.Relationship <> 2 Then
			Universe.ResourcesOfRulers( i ) = Universe.MobilizationFactors( i, 0 ) * ( (
Universe.RPC * Universe.PopDiv( i ) ) / 20000000 ) + Universe.MobilizationFactors( i, 1 )
		Else
			If i == 0 Then
				Universe.ResourcesOfRulers( i ) = Universe.MobilizationFactors( i, 0 ) * (
( Universe.RPC * Universe.SubPopDiv( i ) ) / 20000000 ) + Universe.MobilizationFactors( i, 1 )
			ElseIf Universe.RulerSet( i ) == "c" OR Universe.RulerSet( i ) == "nc" Then
				Universe.ResourcesOfRulers( i ) = Universe.RelationalControl( 1 ) *
Universe.MobilizationFactors( 0, 0 ) * ( ( Universe.RPC * Universe.SubPopDiv( i ) ) / 20000000 )
				Universe.ResourcesOfRulers( 0 ) = Universe.ResourcesOfRulers( 0 ) + (
1 - Universe.RelationalControl( 1 ) ) * Universe.MobilizationFactors( 0, 0 ) * ( ( Universe.RPC *
Universe.SubPopDiv( i ) ) / 20000000 )
			Else
				Universe.ResourcesOfRulers( i ) = Universe.MobilizationFactors( i, 0 ) * (
( Universe.RPC * Universe.PopDiv( i ) ) / 20000000 ) + Universe.MobilizationFactors( i, 1 )
			End If
		End If

		// 潜在的Rulerの顕在化
		If Universe.LatentRulers( i ) == True AND Universe.TerDiv( i ) > 0 Then
			Universe.LatentRulers( i ) = False
			Universe.Counter( 1 ) = Universe.Counter( 1 ) + 1
		End If

		// ウェイト標準化因子の初期化
		Universe.WeightNormalizers( i ) = 0
```

```
                    // 指定Rulerの出力データの構成
                    If i == 0 Then
                            Universe.RulerMonitor( 0, 0 ) = Universe.CellDiv( i )
                            Universe.RulerMonitor( 0, 1 ) = Universe.TerDiv( i )
                            Universe.RulerMonitor( 0, 2 ) = Universe.PopDiv( i )
                            Universe.RulerMonitor( 0, 3 ) = bk( i )
                            If Universe.Relationship == 2 AND Universe.Control1( 13 ) > 0 Then
                                    For j = 1 To Universe.Control1( 13 )
                                            If Universe.RulerSet( j ) <> "s" Then
                                                    Universe.RulerMonitor( 0, 3 ) =
Universe.RulerMonitor( 0, 3 ) + Universe.ResourcesOfRulers( j )
                                            End If
                                    Next j
                            End If
                            Universe.RulerMonitor_Gr( 0 ) = Universe.RulerSet( i )
                    ElseIf i == Universe.NumOfRulers - 1 AND Universe.SpecifiedInsgt <> "" Then
                            Universe.RulerMonitor( 1, 0 ) = Universe.CellDiv( i )
                            Universe.RulerMonitor( 1, 1 ) = Universe.TerDiv( i )
                            Universe.RulerMonitor( 1, 2 ) = Universe.PopDiv( i )
                            Universe.RulerMonitor( 1, 3 ) = bk( i )
                            Universe.RulerMonitor_Gr( 1 ) = Universe.RulerSet( i )
                    End If

                    // 他の出力データの計算
                    If Universe.TerDiv( i ) > 0 Then

                            // 支配領域有するRulerの数のカウント
                            Universe.Counter( 0 ) = Universe.Counter( 0 ) + 1

                            // 領域統治の分裂のエントロピーの計算
                            r = r - Universe.TerDiv( i ) * Log( Universe.TerDiv( i ) )

                            // 主要Rulerの出力データの構成
                            If i == Universe.Govt Then
                                    Universe.MajRulers( 0 ) = CStr( i )
                                    Universe.MajRulers_TerDiv( 0 ) = Universe.TerDiv( i )
                                    Universe.MajRulers_Traits( 0 ) = Universe.TraitsOfRulers( i )
                                    Universe.MajRulers_MblLv( 0 ) = Universe.MobilizationFactors( i, 0 )
                                    Universe.MajRulers_Resources( 0 ) = bk( i )
                                    If Universe.Relationship == 2 AND i == 0 AND Universe.Control1( 13 )
> 0 Then
                                            For j = 1 To Universe.Control1( 13 )
                                                    If Universe.RulerSet( j ) <> "s" Then
                                                            Universe.MajRulers_Resources( 0 ) =
Universe.MajRulers_Resources( 0 ) + Universe.ResourcesOfRulers( j )
                                                    End If
                                            Next j
                                    End If
                                    Universe.MajRulers_Group( 0 ) = Universe.RulerSet( i )
                            Else
                                    For j = 1 To 9
                                            If Universe.TerDiv( i ) > Universe.MajRulers_TerDiv( j ) Then
                                                    If j < 9 Then
                                                            For k = j To 8
                                                                    Universe.MajRulers( 9 + ( j - k ) )
= Universe.MajRulers( 8 + ( j - k ) )
                                                                    Universe.MajRulers_TerDiv( 9 + (
j - k ) ) = Universe.MajRulers_TerDiv( 8 + ( j - k ) )
                                                                    Universe.MajRulers_Traits( 9 + ( j
- k ) ) = Universe.MajRulers_Traits( 8 + ( j - k ) )
```

```
                                                    Universe.MajRulers_MblLv( 9 + ( j
- k ) ) = Universe.MajRulers_MblLv( 8 + ( j - k ) )
                                                    Universe.MajRulers_Resources( 9
+ ( j - k ) ) = Universe.MajRulers_Resources( 8 + ( j - k ) )
                                                    Universe.MajRulers_Group( 9 + ( j
- k ) ) = Universe.MajRulers_Group( 8 + ( j - k ) )
                                                Next k
                                            End If
                                            Universe.MajRulers( j ) = CStr( i )
                                            Universe.MajRulers_TerDiv( j ) = Universe.TerDiv( i )
                                            Universe.MajRulers_Traits( j ) =
Universe.TraitsOfRulers( i )
                                            Universe.MajRulers_MblLv( j ) =
Universe.MobilizationFactors( i, 0 )
                                            Universe.MajRulers_Resources( j ) = bk( i )
                                            If Universe.Relationship == 2 AND i == 0 AND
Universe.Control1( 13 ) > 0 Then
                                                For k = 1 To Universe.Control1( 13 )
                                                    If Universe.RulerSet( k ) <> "s"
Then

Universe.MajRulers_Resources( j ) = Universe.MajRulers_Resources( j ) + Universe.ResourcesOfRulers( k )
                                                    End If
                                                Next k
                                            End If
                                            Universe.MajRulers_Group( j ) = Universe.RulerSet( i
)
                                            break
                                        End If
                                    Next j
                                End If

                            End If

                            // 領域支配率の初期化
                            Universe.TerDiv( i ) = 0

                            // 「同盟」指定がある場合Rulerの集合＝「同盟」を初期化
                            If Universe.Relationship == 1 Then
                                list( i ) = ""
                                Universe.RulerSet( i ) = ""
                            End If

                    Next i
                    Universe.OrderIndex( 0 ) = r / Log( 2 )
                    Universe.OrderIndex( 1 ) = Exp( r )

                    // 「同盟」指定がある場合「主要な脅威」の共有によるRulerの集合＝「同盟」を構成
                    If Universe.Relationship == 1 Then
                        For i = 0 To Universe.Control1( 11 )
                            l_x = Universe.Control1( 9 ) + i
                            For j = 0 To Universe.Control1( 12 )
                                l_y = Universe.Control1( 10 ) + j
                                If Universe.LatticeSpace.Territory( l_x, l_y, 0 ) == 1 Then
                                    k = Universe.LatticeSpace.Ruler( l_x, l_y, 0 )
                                    For l = 0 To 3
                                        n_x = l_x + ( l - 1 ) Mod 2
                                        n_y = l_y + ( l - 2 ) Mod 2
                                        If Universe.LatticeSpace.Territory( n_x, n_y, 0 ) == 1
AND Universe.LatticeSpace.Ruler( n_x, n_y, 0 ) <> k Then
```

```
                                                    m = Universe.LatticeSpace.Ruler( n_x, n_y,
0 )
                                                    str1 = CStr( m )
                                                    If InStr( 1, list( k ), str1 ) == 0 Then
                                                            list( k ) = list( k ) & str1 & ","
                                                            If Universe.RulerSet( k ) == ""
Then
                                                                    If
Universe.ResourcesOfRulers( m ) / Universe.ResourcesOfRulers( k ) > Universe.RelationalControl( 0 ) Then

Universe.RulerSet( k ) = str1
                                                                    End If
                                                            Else
                                                                    n = CInt(
Universe.RulerSet( k ) )
                                                                    If
Universe.ResourcesOfRulers( m ) / Universe.ResourcesOfRulers( k ) > Universe.ResourcesOfRulers( n ) /
Universe.ResourcesOfRulers( k ) Then

Universe.RulerSet( k ) = str1
                                                                    End If
                                                            End If
                                                    End If
                                            End If
                                    Next l
                            End If
                    Next j
            Next i

        // 「分権化」指定がある場合資源の多寡による下位Rulerの分離の有無の決定およびそれに伴う格子点の状
態の更新
        ElseIf Universe.Relationship == 2 Then
            If Universe.Control1( 13 ) > 0 Then
                t = False
                For i = 1 To Universe.Control1( 13 )
                    If Universe.RulerSet( i ) == "nc" AND Universe.ResourcesOfRulers( i ) >
Universe.ResourcesOfRulers( 0 ) Then
                        Universe.RulerSet( i ) = "t"
                        Universe.Counter( 0 ) = Universe.Counter( 0 ) + 1
                        Universe.LatentRulers( i ) = False
                        Universe.Counter( 1 ) = Universe.Counter( 1 ) + 1
                        Universe.SubPopDiv( i ) = 0
                        t = True
                    End If
                Next i
                If t == True Then
                    For i = 0 To Universe.Control1( 11 )
                        l_x = Universe.Control1( 9 ) + i
                        For j = 0 To Universe.Control1( 12 )
                            l_y = Universe.Control1( 10 ) + j
                            If Universe.LatticeSpace.Territory( l_x, l_y, 0 ) == 1
AND Universe.LatticeSpace.Ruler( l_x, l_y, 0 ) == 0 Then
                                l = Universe.LatticeSpace.Region( l_x, l_y, 0
)
                                If l > 0 AND Universe.RulerSet( l ) == "t"
Then
                                    Universe.LatticeSpace.Ruler( l_x,
l_y, 0 ) = l
                                    Universe.LatticeSpace.Region( l_x,
l_y, 0 ) = 0
```

```
                                                                Universe.CellDiv( 0 ) =
Universe.CellDiv( 0 ) - 1
                                                                Universe.CellDiv( l ) =
Universe.CellDiv( l ) + 1
                                                                Universe.PopDiv( 0 ) =
Universe.PopDiv( 0 ) - Universe.LatticeSpace.Pop( l_x, l_y, 0 )
                                                                Universe.PopDiv( l ) =
Universe.PopDiv( l ) + Universe.LatticeSpace.Pop( l_x, l_y, 0 )
                                                        End If

                                                    End If
                                                Next j
                                            Next i
                                            For i = 1 To Universe.Control1( 13 )
                                                If Universe.RulerSet( i ) == "t" Then
                                                    Universe.RulerSet( i ) = "s"
                                                End If
                                            Next i
                                        End If
                                End If

        End If

        // 格子点ごとの処理――状態のチェックとRulerの資源配備の更新、および出力の更新
        For i = 0 To Universe.Control1( 11 )
                l_x = Universe.Control1( 9 ) + i
                For j = 0 To Universe.Control1( 12 )
                        l_y = Universe.Control1( 10 ) + j
                        If Universe.LatticeSpace.Territory( l_x, l_y, 0 ) == 1 Then

                                // 格子点を支配するRulerの取得と領域支配率の計算
                                k = Universe.LatticeSpace.Ruler( l_x, l_y, 0 )
                                Universe.TerDiv( k ) = Universe.TerDiv( k ) +
Universe.LatticeSpace.Area( l_x, l_y, 0 ) / Universe.BasicInfo( 1 )

                                // 前線格子点のチェック
                                If Universe.RulerControl( 4 ) > 0 Then
                                        Universe.LatticeSpace.Frontline( l_x, l_y, 0 ) = 0
                                        For l = 0 To 3
                                                n_x = l_x + ( l - 1 ) Mod 2
                                                n_y = l_y + ( l - 2 ) Mod 2
                                                If Universe.LatticeSpace.Territory( n_x, n_y, 0 ) == 1
AND Universe.LatticeSpace.Ruler( n_x, n_y, 0 ) <> k Then

                                                        // 「同盟」導入の有無で前線判断類別
                                                        If Universe.Relationship <> 1 Then
                                                                Universe.LatticeSpace.Frontline(
l_x, l_y, 0 ) = 1

                                                                Break
                                                        ElseIf Universe.Relationship == 1 Then
                                                                m = Universe.LatticeSpace.Ruler(
n_x, n_y, 0 )

                                                                If Universe.RulerSet( k ) == "" OR
Universe.RulerSet( m ) <> Universe.RulerSet( k ) Then

Universe.LatticeSpace.Frontline( l_x, l_y, 0 ) = 1
                                                                        Break
                                                                End If
                                                        End If

                                                End If
```

```
                                        Next l
                                End If

                                // 資源配備更新とウェイト標準化因子の計算
                                Universe.LatticeSpace.Deployment( l_x, l_y, 0 ) = CStr( k ) & "," & CStr(
( 1 + Universe.RulerControl( 4 ) * Universe.LatticeSpace.Frontline( l_x, l_y, 0 ) ) * Log(
Universe.LatticeSpace.Pop( l_x, l_y, 0 ) ) * Universe.ResourcesOfRulers( k ) )
                                Universe.WeightNormalizers( k ) = Universe.WeightNormalizers( k ) + (
1 + Universe.RulerControl( 4 ) * Universe.LatticeSpace.Frontline( l_x, l_y, 0 ) ) * Log(
Universe.LatticeSpace.Pop( l_x, l_y, 0 ) )

                                // 「分権化」指定の場合下位Rul-
erによる資源配備の追加とウェイト標準化因子の計算
                                If Universe.Relationship == 2 AND k == 0 Then
                                        l = Universe.LatticeSpace.Region( l_x, l_y, 0 )
                                        If l > 0 Then
                                                Universe.LatticeSpace.Deployment( l_x, l_y, 0 ) =
Universe.LatticeSpace.Deployment( l_x, l_y, 0 ) & "," & CStr( ( 1 + Universe.RulerControl( 4 ) *
Universe.LatticeSpace.Frontline( l_x, l_y, 0 ) ) * Log( Universe.LatticeSpace.Pop( l_x, l_y, 0 ) ) *
Universe.ResourcesOfRulers( l ) )
                                                Universe.WeightNormalizers( l ) =
Universe.WeightNormalizers( l ) + ( 1 + Universe.RulerControl( 4 ) * Universe.LatticeSpace.Frontline( l_x, l_y,
0 ) ) * Log( Universe.LatticeSpace.Pop( l_x, l_y, 0 ) )
                                        End If
                                End If

                                // Ruler表示色フルカラー化のための出力処理
                                If k <> 0 AND ( l_x <> Universe.Capital( 0, 0 ) OR l_y <>
Universe.Capital( 0, 1 ) ) Then
                                        one = CreateAgt( Universe.LatticeSpace.Display )
                                        one.X = l_x
                                        one.Y = l_y
                                        If k == Universe.NumOfRulers - 1 AND
Universe.SpecifiedInsgt <> "" Then
                                                one.Color = Color_Blue
                                        Else
                                                one.Color = Color_White * CDbl( k ) / CDbl(
Universe.NumOfRulers )
                                        End If
                                End If

                        End If
                Next j
        Next i

        // データの書き出し
        If Universe.FilesOut == True Then
                str1 = "Output//" & Universe.Country & "//dynamics.txt"
                str2 = CStr( GetCountSimulationNumber( ) ) & " " & CStr( GetCountStep( ) ) & " " & CStr(
Universe.OrderIndex( 0 ) ) & " " & CStr( Universe.OrderIndex( 1 ) ) & " " & CStr( Universe.Counter( 0 ) ) & " "
& CStr( Universe.Counter( 1 ) ) & " " & CStr( CBool( Universe.RulerMonitor( 0, 1 ) ) ) & " " & CStr( CBool(
Universe.RulerMonitor( 1, 1 ) ) ) & " "
                For i = 0 To 1
                        For j = 0 To 3
                                str2 = str2 & CStr( Universe.RulerMonitor( i, j ) ) & " "
                        Next j
                        If Universe.RulerMonitor_Gr( i ) <> "" Then
                                str2 = str2 & Universe.RulerMonitor_Gr( i ) & " "
                        Else
                                str2 = str2 & "void "
                        End If
```

```
                Next i
                For i = 0 To 9
                        If Universe.MajRulers( i ) == "" Then
                                str2 = str2 & "void " & "void " & CStr( Universe.MajRulers_MblLv( i ) ) &
" " & CStr( Universe.MajRulers_TerDiv( i ) ) & " " & CStr( Universe.MajRulers_Resources( i ) ) & " void "
                        ElseIf Universe.MajRulers_Group( i ) == "" Then
                                str2 = str2 & Universe.MajRulers( i ) & " " & Universe.MajRulers_Traits(
i ) & " " & CStr( Universe.MajRulers_MblLv( i ) ) & " " & CStr( Universe.MajRulers_TerDiv( i ) ) & " " & CStr(
Universe.MajRulers_Resources( i ) ) & " void "
                        Else
                                str2 = str2 & Universe.MajRulers( i ) & " " & Universe.MajRulers_Traits(
i ) & " " & CStr( Universe.MajRulers_MblLv( i ) ) & " " & CStr( Universe.MajRulers_TerDiv( i ) ) & " " & CStr(
Universe.MajRulers_Resources( i ) ) & " " & Universe.MajRulers_Group( i ) & " "
                        End If
                Next i
                str2 = str2 & CStr( Universe.GrossPotential ) & " "
                OpenFile( str1, 1, 3 )
                WriteLnFile( 1, str2 )
                CloseFile( 1 )
        End If

        // 指定の場合画面撮影
        If Universe.Snapshot == True AND ( GetCountStep( ) <= 20 OR GetCountStep( ) Mod 10 == 0 ) Then
                str1 = "Output//" & Universe.Country & "//Pics//" & TimeToStr( GetRealTime( ),
"yyyyMMddHHmmss_" ) & CStr( GetCountSimulationNumber( ) ) & "_" & CStr( GetCountStep( ) ) & ".jpg"
                ScreenShot( str1 )
        End If

        // シミュレーションの終了処理
        If GetCountStep( ) == Universe.TimeManager Then
                If Universe.FilesOut == True Then
                        str1 = "Output//" & Universe.Country & "//pic" & CStr( GetCountSimulationNum-
ber( ) ) & ".jpg"
                        ScreenShot( str1 )
                End If
                ExitSimulation( )
        End If

        // マップのクリック情報の出力
        If IsMouseClickedOnMap( ) == True Then
                ClearConsoleScreen( )
                GetClickedMapPosition( l_x, l_y, str1 )
                If Universe.LatticeSpace.Territory( l_x, l_y, 0 ) == 0 Then
                        PrintLn( "Careful! Off the territory!" )
                Else
                        PrintLn( "******** PopCell Information ********" )
                        str1 = "Point : " & "( " & CStr( l_x ) & "," & CStr( l_y ) & " )"
                        If l_x == Universe.Capital( 0, 0 ) AND l_y == Universe.Capital( 0, 1 ) Then
                                str1 = str1 & " --capital--"
                        End If
                        PrintLn( str1 )
                        str1 = "Area : " & CStr( Universe.LatticeSpace.Area( l_x, l_y, 0 ) ) & " km2"
                        PrintLn( str1 )
                        str1 = "Population : " & CStr( CInt( Universe.LatticeSpace.Pop( l_x, l_y, 0 ) ) )
                        PrintLn( str1 )
                        k = 0
                        For i = 0 To Universe.Control1( 1 ) - 1
                                str1 = "Trait" & CStr( 1 + i ) & " : "
                                For j = 0 To Universe.Control1( 6 + i ) - 1
                                        str2 = GetToken( Universe.LatticeSpace.Traits( l_x, l_y, 0 ), k +
j )
```

```
                                        If str2 <> "0" Then
                                                str1 = str1 & str2 & "(" & GetToken(
Universe.LatticeSpace.Comp( l_x, l_y, 0 ), k + j ) & ") "
                                        End If
                                Next j
                                PrintLn( str1 )
                                k = k + Universe.Control1( 6 + i )
                        Next i
                        k = Universe.LatticeSpace.Ruler( l_x, l_y, 0 )
                        str1 = "Ruler : " & CStr( k )
                        PrintLn( str1 )
                        If Universe.Relationship == 2 AND k == 0 Then
                                l = Universe.LatticeSpace.Region( l_x, l_y, 0 )
                                str1 = "Region : " & CStr( l )
                                PrintLn( str1 )
                        End If
                        e = 1 - CDbl( GetToken( Universe.LatticeSpace.CDistance( l_x, l_y, 0 ), k ) )
                        If Universe.Relationship == 2 AND k == 0 AND l > 0 Then
                                e = ( 1 - Universe.RelationalControl( 1 ) ) * ( 1 - CDbl( GetToken(
Universe.LatticeSpace.CDistance( l_x, l_y, 0 ), k ) ) ) + Universe.RelationalControl( 1 ) * ( 1 - CDbl( GetToken(
Universe.LatticeSpace.CDistance( l_x, l_y, 0 ), l ) ) )
                        End If
                        str1 = "Fit with Ruler : " & CStr( e )
                        PrintLn( str1 )
                        str1 = "Deployment : "
                        If CInt( GetToken( Universe.LatticeSpace.Deployment( l_x, l_y, 0 ), 0 ) ) == k
Then
                                If Universe.WeightNormalizers( k ) <= 0 Then
                                        str1 = str1 & CStr( Universe.ResourcesOfRulers( k ) )
                                Else
                                        str1 = str1 & CStr( CDbl( GetToken(
Universe.LatticeSpace.Deployment( l_x, l_y, 0 ), 1 ) ) / Universe.WeightNormalizers( k ) )
                                End If
                                If Universe.Relationship == 2 AND k == 0 AND l > 0 Then
                                        If Universe.WeightNormalizers( l ) <= 0 Then
                                                str1 = str1 & "+" & CStr(
Universe.ResourcesOfRulers( l ) )
                                        Else
                                                str1 = str1 & "+" & CStr( CDbl( GetToken(
Universe.LatticeSpace.Deployment( l_x, l_y, 0 ), 2 ) ) / Universe.WeightNormalizers( l ) )
                                        End If
                                End If
                        Else
                                str1 = str1 & "0.0"
                        End If
                        PrintLn( str1 )
                        PrintLn( "***********************************" )
                        PrintLn( "" )
                        PrintLn( "********* Ruler Information *********" )
                        str1 = "ID : " & CStr( k )
                        PrintLn( str1 )
                        If k == Universe.Govt Then
                                PrintLn( "Status : Government" )
                        Else
                                PrintLn( "Status : Insurgent" )
                        End If
                        str1 = "Point of Entry : " &  "( " & CStr( Universe.Capital( k, 0 ) ) & "," & CStr(
Universe.Capital( k, 1 ) ) & " )"
                        PrintLn( str1 )
                        str2 = Universe.TraitsOfRulers( k )
                        str2 = Left( str2, Len( str2 ) - 1 )
```

```
                        str1 = "Traits : " & str2
                        PrintLn( str1 )
                        str1 = "Mobilization Factors : " & CStr( Universe.MobilizationFactors( k, 0 ) ) &
"(endo.)  " & CStr( Universe.MobilizationFactors( k, 1 ) ) & "(exo.)"
                        PrintLn( str1 )
                        str1 = "Territory : " & CStr( Universe.TerDiv( k ) * 100 ) & " %"
                        PrintLn( str1 )
                        str1 = "Population : " & CStr( CInt( Universe.PopDiv( k ) ) )
                        PrintLn( str1 )
                        r = Universe.ResourcesOfRulers( k )
                        If Universe.Relationship == 2 AND k == 0 AND Universe.Control1( 13 ) > 0 Then
                                For i = 1 To Universe.Control1( 13 )
                                        If Universe.RulerSet( i ) <> "s" Then
                                                r = r + Universe.ResourcesOfRulers( i )
                                        End If
                                Next i
                        End If
                        str1 = "Resources : " & CStr( r )
                        PrintLn( str1 )
                        str1 = "Group : " & Universe.RulerSet( k )
                        PrintLn( str1 )
                        PrintLn( "**********************************" )
                        If Universe.Relationship == 2 AND k == 0 AND l > 0 Then
                                PrintLn( "" )
                                PrintLn( "********* SubRuler Information *********" )
                                str1 = "Region : " & CStr( l )
                                PrintLn( str1 )
                                If Universe.RulerSet( l ) == "c" Then
                                        PrintLn( "Status : Core Region" )
                                Else
                                        PrintLn( "Status : Non-Core Region" )
                                End If
                                str1 = "Point of Entry : " &  "( " & CStr( Universe.Capital( l, 0 ) ) & "," &
CStr( Universe.Capital( l, 1 ) ) & " )"
                                PrintLn( str1 )
                                str2 = Universe.TraitsOfRulers( l )
                                str2 = Left( str2, Len( str2 ) - 1 )
                                str1 = "Traits : " & str2
                                PrintLn( str1 )
                                str1 = "Population : " & CStr( CInt( Universe.SubPopDiv( l ) ) )
                                PrintLn( str1 )
                                str1 = "Resources : " & CStr( Universe.ResourcesOfRulers( l ) )
                                PrintLn( str1 )
                                PrintLn( "**********************************" )
                        End If
                End If
        End If

}

Univ_Finish{

}

// ギブスサンプラーによる状態更新のアルゴリズム
Sub GibbsSampler( t_x As Integer, t_y As Integer )
{

        Dim i As Integer
        Dim j As Integer
        Dim k As Integer
```

```
	Dim n_x As Integer
	Dim n_y As Integer
	Dim H( 1000 ) As Double
	Dim B( 1000 ) As Double
	Dim Z As Double
	Dim r As Double
	Dim t As Boolean

	// ギブスサンプラーにしたがって各Rulerのもとでのポテンシャルの計算
	Z = 0
	For i = 0 To Universe.NumOfRulers - 1
		If Universe.Relationship <> 2 OR ( Universe.RulerSet( i ) == "" OR Universe.RulerSet( i ) ==
"s" ) Then
			H( i ) = 0

			// 潜在的Rulerの場合一点のみに影響が及ぶ
			If Universe.LatentRulers( i ) == True AND ( t_x == Universe.Capital( i, 0 ) AND
t_y == Universe.Capital( i, 1 ) ) Then
				H( i ) = H( i ) - ( 1 - CDbl( GetToken( Universe.LatticeSpace.CDistance(
t_x, t_y, 0 ), i ) ) ) - Universe.PotentialControl( 1 ) * ( 1 - Universe.MobilizationFactors( i, 0 ) ) -
Universe.PotentialControl( 0 ) * Universe.ResourcesOfRulers( i )

			// それ以外の場合
			Else
				t = False

				// 近傍Rulerかのチェックと多体ポテンシャル＝Rulerの強制力行使の計算
				If CInt( GetToken( Universe.LatticeSpace.Deployment( t_x, t_y, 0 ), 0 ) )
== i Then
					t = True
					If Universe.WeightNormalizers( i ) <= 0 Then
						H( i ) = H( i ) - Universe.PotentialControl( 0 ) *
Universe.ResourcesOfRulers( i )
					Else
						H( i ) = H( i ) - Universe.PotentialControl( 0 ) * (
CDbl( GetToken( Universe.LatticeSpace.Deployment( t_x, t_y, 0 ), 1 ) ) / Universe.WeightNormalizers( i ) )
					End If

					// 「分権化」指定の場合下位Rulerからのポテンシャルも加算
					If Universe.Relationship == 2 AND i == 0 Then
						k = Universe.LatticeSpace.Region( t_x, t_y, 0 )
						If k > 0 Then
							If Universe.WeightNormalizers( k ) <= 0
Then
								H( i ) = H( i ) -
Universe.PotentialControl( 0 ) * Universe.ResourcesOfRulers( k )
							Else
								H( i ) = H( i ) -
Universe.PotentialControl( 0 ) * ( CDbl( GetToken( Universe.LatticeSpace.Deployment( t_x, t_y, 0 ), 2 ) ) /
Universe.WeightNormalizers( k ) )
							End If
						End If
					End If

				End If
				For j = 0 To 3
					n_x = t_x + ( j - 1 ) Mod 2
					n_y = t_y + ( j - 2 ) Mod 2
					If Universe.LatticeSpace.Territory( n_x, n_y, 0 ) == 1 AND
CInt( GetToken( Universe.LatticeSpace.Deployment( n_x, n_y, 0 ), 0 ) ) == i Then
						t = True
```

```
                                                If Universe.WeightNormalizers( i ) <= 0 Then
                                                        H( i ) = H( i ) - Universe.PotentialControl( 0
) * Universe.ResourcesOfRulers( i )
                                                Else
                                                        H( i ) = H( i ) - Universe.PotentialControl( 0
) * ( CDbl( GetToken( Universe.LatticeSpace.Deployment( n_x, n_y, 0 ), 1 ) ) / Universe.WeightNormalizers( i )
)
                                                End If

                                                // 「分権化」指定の場合下位Rul-
erからのポテンシャルも加算
                                                If Universe.Relationship == 2 AND i == 0 Then
                                                        k = Universe.LatticeSpace.Region( n_x, n_y,
0 )
                                                        If k > 0 Then
                                                                If Universe.WeightNormalizers( k
) <= 0 Then
                                                                        H( i ) = H( i ) -
Universe.PotentialControl( 0 ) * Universe.ResourcesOfRulers( k )
                                                                Else
                                                                        H( i ) = H( i ) -
Universe.PotentialControl( 0 ) * ( CDbl( GetToken( Universe.LatticeSpace.Deployment( n_x, n_y, 0 ), 2 ) ) /
Universe.WeightNormalizers( k ) )
                                                                End If
                                                        End If
                                                End If

                                        End If

                                        // 「同盟」指定の場合「盟友」Rulerのポテンシャルも算入
                                        If Universe.Relationship == 1 AND
Universe.LatticeSpace.Territory( n_x, n_y, 0 ) == 1 AND CInt( GetToken( Universe.LatticeSpace.Deployment(
n_x, n_y, 0 ), 0 ) ) <> i Then
                                                k = CInt( GetToken(
Universe.LatticeSpace.Deployment( n_x, n_y, 0 ), 0 ) )
                                                If Universe.RulerSet( i ) <> "" AND Universe.RulerSet(
k ) == Universe.RulerSet( i ) Then
                                                        If Universe.WeightNormalizers( k ) <= 0
Then
                                                                H( i ) = H( i ) -
Universe.PotentialControl( 0 ) * Universe.ResourcesOfRulers( k )
                                                        Else
                                                                H( i ) = H( i ) -
Universe.PotentialControl( 0 ) * ( CDbl( GetToken( Universe.LatticeSpace.Deployment( n_x, n_y, 0 ), 1 ) ) /
Universe.WeightNormalizers( k ) )
                                                        End If
                                                End If
                                        End If

                                Next j

                                // 近傍Rulerの場合は 1 体ポテンシャル＝住民の近傍Rul-
erへのロイヤリティの計算
                                If Universe.LatentRulers( i ) == False AND t == True Then
                                        If Universe.Relationship <> 2 Then
                                                H( i ) = H( i ) - ( 1 - CDbl( GetToken(
Universe.LatticeSpace.CDistance( t_x, t_y, 0 ), i ) ) ) - Universe.PotentialControl( 1 ) * ( 1 -
Universe.MobilizationFactors( i, 0 ) )

                                        // 「分権化」指定の場合は下位Rulerへのロイヤリティも反映
                                        Else
```

```
                                                k = Universe.LatticeSpace.Region( t_x, t_y, 0 )
                                                If i == 0 AND k > 0 Then
                                                        H( i ) = H( i ) - ( 1 -
Universe.RelationalControl( 1 ) ) * ( 1 - CDbl( GetToken( Universe.LatticeSpace.CDistance( t_x, t_y, 0 ), i ) ) ) -
Universe.RelationalControl( 1 ) * ( 1 - CDbl( GetToken( Universe.LatticeSpace.CDistance( t_x, t_y, 0 ), k ) ) ) -
Universe.PotentialControl( 1 ) * ( 1 - Universe.MobilizationFactors( i, 0 ) )
                                                Else
                                                        H( i ) = H( i ) - ( 1 - CDbl( GetToken(
Universe.LatticeSpace.CDistance( t_x, t_y, 0 ), i ) ) ) - Universe.PotentialControl( 1 ) * ( 1 -
Universe.MobilizationFactors( i, 0 ) )
                                                End If
                                        End If

                                        // 「同盟」指定の場合、近接Rulerでももとの Ruler
の「盟友」ならポテンシャルの壁
                                        If Universe.Relationship == 1 Then
                                                k = CInt( GetToken(
Universe.LatticeSpace.Deployment( t_x, t_y, 0 ), 0 ) )
                                                If i <> k AND Universe.RulerSet( i ) <> "" AND
Universe.RulerSet( i ) == Universe.RulerSet( k ) Then
                                                        H( i ) = Universe.PotentialControl( 3 )
                                                End If
                                        End If

                                // それ以外の場合はポテンシャルの壁により事実上状態更新の対象外
                                Else
                                        H( i ) = Universe.PotentialControl( 2 )

                                End If

                        End If

                        // ポテンシャルをもとにRuler i の支配下にある確率のボルツマン因子を計算
                        B( i ) = Exp( -1 * ( 1 / Universe.Noise ) * H( i ) )
                        Z = Z + B( i )

                // 「分権化」指定の場合下位Rulerは状態更新の対象外
                Else
                        B( i ) = 0

                End If
        Next i

        // 状態更新
        r = Rnd( )
        For i = 0 To Universe.NumOfRulers - 1
                r = r - B( i ) / Z
                If r < 0 Then
                        j = Universe.LatticeSpace.Ruler( t_x, t_y, 0 )
                        Universe.LatticeSpace.Ruler( t_x, t_y, 0 ) = i
                        Universe.LatticeSpace.Color2( t_x, t_y, 0 ) = 0.2 + 0.65 * CDbl(
Universe.NumOfRulers - i ) / CDbl( Universe.NumOfRulers )

                        // 首都が状態変更する場合の処理
                        If t_x == Universe.Capital( 0, 0 ) AND t_y == Universe.Capital( 0, 1 ) Then
                                Universe.Govt = i
                                Universe.LatticeSpace.Color2( t_x, t_y, 0 ) = 1
                        End If

                        // 状態更新に伴うセル・領域・人口分割の更新
                        Universe.CellDiv( i ) = Universe.CellDiv( i ) + 1
```

```
                        Universe.TerDiv( i ) = Universe.TerDiv( i ) + Universe.LatticeSpace.Area( t_x, t_y,
0 ) / Universe.BasicInfo( 1 )
                        Universe.PopDiv( i ) = Universe.PopDiv( i ) + Universe.LatticeSpace.Pop( t_x, t_y,
0 )
                        Universe.CellDiv( j ) = Universe.CellDiv( j ) - 1
                        Universe.TerDiv( j ) = Universe.TerDiv( j ) - Universe.LatticeSpace.Area( t_x, t_y,
0 ) / Universe.BasicInfo( 1 )
                        If Universe.TerDiv( j ) < 1 / ( 10 ^ 10 ) Then
                                Universe.TerDiv( j ) = 0
                        End If
                        Universe.PopDiv( j ) = Universe.PopDiv( j ) - Universe.LatticeSpace.Pop( t_x, t_y,
0 )

                        // 「分権化」指定の場合下位Rulerの人口構成も更新
                        If Universe.Relationship == 2 Then
                                k = Universe.LatticeSpace.Region( t_x, t_y, 0 )
                                If i == 0 Then
                                        Universe.SubPopDiv( k ) = Universe.SubPopDiv( k ) +
Universe.LatticeSpace.Pop( t_x, t_y, 0 )
                                End If
                                If j == 0 Then
                                        Universe.SubPopDiv( k ) = Universe.SubPopDiv( k ) -
Universe.LatticeSpace.Pop( t_x, t_y, 0 )
                                End If
                        End If

                        // 総ポテンシャルの加算
                        Universe.GrossPotential = Universe.GrossPotential + H( i )

                        break
                End If
        Next i

}

// 住民とRulerの属性ベクトル間の正規化された「距離」を求める関数
Function StdCedermanDistance( tr1 As String, tr2 As String, com As String ) As Double
{

        Dim i As Integer
        Dim j As Integer
        Dim k As Integer
        Dim l As Integer
        Dim m As Integer
        Dim d As Double
        Dim e As Double
        Dim w As Double
        Dim rtn As Double
        Dim str1 As String
        Dim str2 As String

        rtn = 0
        l = 0
        For i = 0 To Universe.Control1( 1 ) - 1
                d = 0
                m = Universe.Control2( 1 + i ) / 2
                str2 = GetToken( tr2, i )
                For j = 0 To Universe.Control1( 6 + i ) - 1
                        str1 = GetToken( tr1, l + j )
                        w = CDbl( GetToken( com, l + j ) )
                        e = 0
```

```
                        If str1 <> "" Then
                            For k = 0 To m - 1
                                If Mid( str2, 2 * k + 1, 2 ) == "00" Then
                                    break
                                ElseIf Mid( str2, 2 * k + 1, 2 ) == "**" Then
                                    e = 0.5
                                    break
                                ElseIf Mid( str1, 2 * k + 1, 2 ) <> Mid( str2, 2 * k + 1, 2 ) Then
                                    e = 1
                                    break
                                End If
                            Next k
                        End If
                        d = d + w * e
                    Next j
                    rtn = rtn + d
                    l = l + Universe.Control1( 6 + i )
                Next i
                rtn = rtn / CDbl( Universe.Control1( 1 ) )
                Return( rtn )

}
#end_rule UNIVERSE




//---------------------------------------------------
// Agent Rule
//---------------------------------------------------
#begin_rule UNIVERSE.LATTICESPACE.DISPLAY
Agt_Init{

}

Agt_Step{

        // 出力更新終われば消去
        DelAgt( My )

}
#end_rule UNIVERSE.LATTICESPACE.DISPLAY




//---------------------------------------------------
// Simulation Definition Division
//---------------------------------------------------
Single_Simulation {
    Step_Max = 0;
    Time_Max = 0;
    End_Condition = "";
    Exec_Wait = 0;
    Exec_Order = Random;
    Exec_Order_Random_First_Step_Only = No;
    Random_Seed = 0;
    Log_File = "", Fixed, Append, 0;
    Replay_Log_File = "", Fixed, Append, 0;
    Redraw_Timing = Step;
    Redraw_Interval = 1;
```

```
    Garbage_Interval = 0;
}


//---------------------------------------------------
// Simulation Loop Definition Division
//---------------------------------------------------
Repeat_Simulation {
    Value_Change = None;
    Initial_Value_Change_None {
        Repeat_Max = 1;
    }
}


//---------------------------------------------------
// 2 Dimension Map Display Definition Division
//---------------------------------------------------
Map_Output {
    Map_Space_Name = "UNIVERSE.LatticeSpace";
    Map_Name = "TraitsConfig";
    Font_Name = "Dialog";
    Font_Char_Set = 81;
    Font_Size = 12.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 255, 255, 255;
    Title = "";
    Axis_Label = "", "";
    Draw_Range = 0, 0, 49, 49;
    Ruled_Line = No;
    Chess_Type_Display_Position = No;
    Remarks = No;
    Legend_Pos = Top;
    Background_Color = 0, 0, 0;
    Background_Transparent = No;
    Target_Layer = 0;
    Coordinate_Display = No;
    Space_Kind = Square_2D;
    BG_Pict = No;
    BG_Type = 0;
    BG_Var_Name = "";
    BG_File_Name = "";
    Disable = Yes;
    Position_x = 248;
    Position_y = 20;
    Size_x = 682;
    Size_y = 585;
    Origin_Position = 1;
    Output {
        Map_Element_Name = "Color1";
        Map_Element = Space_Variable;
        Output_Expression = "UNIVERSE.LatticeSpace.Color1";
        Space_Draw_Range = 0.0, 1.0;
        Space_Color_Min = 240, 240, 255;
        Space_Color_Max = 0, 0, 102;
        Marker_Id = -1;
        Icon_Type = 0;
        Icon_Var_Name = "";
        Icon_Transparent = No;
        Icon_Enlarg_Reduce = No;
```

```
        Icon_File_Name = "";
        Num_Display = No;
        Num_Var_Name = "";
    }
}


//---------------------------------------------------
// 2 Dimension Map Display Definition Division
//---------------------------------------------------
Map_Output {
    Map_Space_Name = "UNIVERSE.LatticeSpace";
    Map_Name = "RulerConfig";
    Font_Name = "Dialog";
    Font_Char_Set = 81;
    Font_Size = 12.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 255, 255, 255;
    Title = "";
    Axis_Label = "", "";
    Draw_Range = 0, 0, 49, 49;
    Ruled_Line = No;
    Chess_Type_Display_Position = No;
    Remarks = No;
    Legend_Pos = Top;
    Background_Color = 0, 0, 0;
    Background_Transparent = No;
    Target_Layer = 0;
    Coordinate_Display = No;
    Space_Kind = Square_2D;
    BG_Pict = No;
    BG_Type = 0;
    BG_Var_Name = "";
    BG_File_Name = "";
    Disable = No;
    Position_x = 172;
    Position_y = 2;
    Size_x = 630;
    Size_y = 544;
    Origin_Position = 1;
    Output {
        Map_Element_Name = "Display";
        Map_Element = Agent_Variable;
        Output_Expression = "UNIVERSE.LatticeSpace.Display";
        Draw_Line = No;
        Collection_Name = "";
        Line_Type = 0;
        Line_Arrow = 0;
        Line_Width = 0;
        Line_Color = 0, 0, 0;
        Line_Type_Variable = "";
        Line_Arrow_Variable = "";
        Line_Width_Variable = "";
        Line_Color_Variable = "";
        Agent_Color = "UNIVERSE.LatticeSpace.Display.Color";
        Marker_Id = 2;
        Effective_Figures = 0;
        Format_Id = 0;
        Icon_Type = 0;
        Icon_Var_Name = "";
```

```
        Icon_Transparent = No;
        Icon_Enlarg_Reduce = No;
        Icon_File_Name = "";
        Num_Display = No;
        Num_Var_Name = "";
    }
    Output {
        Map_Element_Name = "Color2";
        Map_Element = Space_Variable;
        Output_Expression = "UNIVERSE.LatticeSpace.Color2";
        Space_Draw_Range = 0.0, 1.0;
        Space_Color_Min = 255, 255, 215;
        Space_Color_Max = 204, 0, 0;
        Marker_Id = -1;
        Icon_Type = 0;
        Icon_Var_Name = "";
        Icon_Transparent = No;
        Icon_Enlarg_Reduce = No;
        Icon_File_Name = "";
        Num_Display = No;
        Num_Var_Name = "";
    }
}


//---------------------------------------------------
// Numeric Data Window Output Definition Division
//---------------------------------------------------
Numeric_Output {
    Numeric_Name = "Output";
    Font_Name = "Dialog";
    Font_Char_Set = 81;
    Font_Size = 12.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 255, 255, 255;
    Title = "";
    Disable = No;
    Position_x = 803;
    Position_y = 3;
    Size_x = 242;
    Size_y = 580;
    Output {
        Output_Element_Name = "Simulation";
        Figures = 0;
        Format_Id = 0;
        Output_Expression = "GetCountSimulationNumber( )";
    }
    Output {
        Output_Element_Name = "Period";
        Figures = 0;
        Format_Id = 0;
        Output_Expression = "GetCountStep( )";
    }
    Output {
        Output_Element_Name = "Entropy";
        Figures = 3;
        Format_Id = 0;
        Output_Expression = "Universe.OrderIndex( 0 )";
    }
    Output {
```

```
    Output_Element_Name = "EN of Rulers";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.OrderIndex( 1 )";
}
Output {
    Output_Element_Name = "Num of Present Rulers";
    Figures = 0;
    Format_Id = 0;
    Output_Expression = "Universe.Counter( 0 )";
}
Output {
    Output_Element_Name = "Num of Emergent Rulers";
    Figures = 0;
    Format_Id = 0;
    Output_Expression = "Universe.Counter( 1 )";
}
Output {
    Output_Element_Name = "InitGovt. Territory";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.RulerMonitor( 0, 1 )";
}
Output {
    Output_Element_Name = "SInsgt. Territory";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.RulerMonitor( 1, 1 )";
}
Output {
    Output_Element_Name = "Govt. Traits";
    Figures = 0;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_Traits( 0 )";
}
Output {
    Output_Element_Name = "Govt. Mobilization";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_MblLv( 0 )";
}
Output {
    Output_Element_Name = "Govt. Teirritory";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_TerDiv( 0 )";
}
Output {
    Output_Element_Name = "Govt. Group";
    Figures = 0;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_Group( 0 )";
}
Output {
    Output_Element_Name = "Insgt.1 Traits";
    Figures = 0;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_Traits( 1 )";
}
Output {
    Output_Element_Name = "Insgt.1 Mobilization";
```

```
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_MblLv( 1 )";
}
Output {
    Output_Element_Name = "Insgt.1 Territory";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_TerDiv( 1 )";
}
Output {
    Output_Element_Name = "Insgt.1 Group";
    Figures = 0;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_Group( 1 )";
}
Output {
    Output_Element_Name = "Insgt.2 Traits";
    Figures = 0;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_Traits( 2 )";
}
Output {
    Output_Element_Name = "Insgt.2 Mobilization";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_MblLv( 2 )";
}
Output {
    Output_Element_Name = "Insgt.2 Territory";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_TerDiv( 2 )";
}
Output {
    Output_Element_Name = "Insgt.2 Group";
    Figures = 0;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_Group( 2 )";
}
Output {
    Output_Element_Name = "Insgt.3 Traits";
    Figures = 0;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_Traits( 3 )";
}
Output {
    Output_Element_Name = "Insgt.3 Mobilization";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_MblLv( 3 )";
}
Output {
    Output_Element_Name = "Insgt.3 Territory";
    Figures = 3;
    Format_Id = 0;
    Output_Expression = "Universe.MajRulers_TerDiv( 3 )";
}
Output {
    Output_Element_Name = "Insgt.3 Group";
    Figures = 0;
```

```
        Format_Id = 0;
        Output_Expression = "Universe.MajRulers_Group( 3 )";
    }
    Output {
        Output_Element_Name = "Num of Cells";
        Figures = 0;
        Format_Id = 0;
        Output_Expression = "Universe.BasicInfo( 0 )";
    }
    Output {
        Output_Element_Name = "Area";
        Figures = 2;
        Format_Id = 0;
        Output_Expression = "Universe.BasicInfo( 1 )";
    }
    Output {
        Output_Element_Name = "Pop";
        Figures = 0;
        Format_Id = 0;
        Output_Expression = "Universe.BasicInfo( 2 )";
    }
    Output {
        Output_Element_Name = "PerCapitaResources";
        Figures = 3;
        Format_Id = 0;
        Output_Expression = "Universe.RPC";
    }
    Output {
        Output_Element_Name = "Types in Traits1";
        Figures = 0;
        Format_Id = 0;
        Output_Expression = "Universe.BasicInfo( 3 )";
    }
    Output {
        Output_Element_Name = "Types in Traits2";
        Figures = 0;
        Format_Id = 0;
        Output_Expression = "Universe.BasicInfo( 4 )";
    }
    Output {
        Output_Element_Name = "Types in Traits3";
        Figures = 0;
        Format_Id = 0;
        Output_Expression = "Universe.BasicInfo( 5 )";
    }
    Output {
        Output_Element_Name = "Gross Potential";
        Figures = 3;
        Format_Id = 0;
        Output_Expression = "Universe.GrossPotential";
    }
}


//-------------------------------------------------------
// Control Panel Definition Division
//-------------------------------------------------------
Control_Panel {
    Display_Area = 1113, 599, 517, 381;
    Panel_Item {
        Item_Type = Edit_Box;
```

```
    Item_Name = "Country";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "UNIVERSE.Country";
    Text_Data = "ETH";
  }
  Panel_Item {
    Item_Type = Edit_Box;
    Item_Name = "Govt. Traits";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "UNIVERSE.TraitsOfRulers(0)";
    Text_Data = "";
  }
  Panel_Item {
    Item_Type = Edit_Box;
    Item_Name = "Specified Insgt.";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "UNIVERSE.SpecifiedInsgt";
    Text_Data = "";
  }
  Panel_Item {
    Item_Type = Toggle;
    Item_Name = "Whole Data In";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "UNIVERSE.Control2(0)";
    Control_On = Yes;
    Control_On_Int = 1;
    Control_On_Long = 0;
    Control_On_Double = 1.0;
    Control_On_String = "";
    Control_Off = No;
    Control_Off_Int = 0;
    Control_Off_Long = 0;
    Control_Off_Double = 0.0;
    Control_Off_String = "";
    Toggle_Button_On = No;
    Shortcut_Key = "";
  }
```

```
    Panel_Item {
        Item_Type = Toggle;
        Item_Name = "Data Files Out";
        Display_Area = 0, 0, 0, 0;
        Font_Name = "";
        Font_Char_Set = 0;
        Font_Size = 0.0;
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "UNIVERSE.FilesOut";
        Control_On = Yes;
        Control_On_Int = 1;
        Control_On_Long = 0;
        Control_On_Double = 1.0;
        Control_On_String = "";
        Control_Off = No;
        Control_Off_Int = 0;
        Control_Off_Long = 0;
        Control_Off_Double = 0.0;
        Control_Off_String = "";
        Toggle_Button_On = Yes;
        Shortcut_Key = "";
    }
    Panel_Item {
        Item_Type = Toggle;
        Item_Name = "Snapshot";
        Display_Area = 0, 0, 0, 0;
        Font_Name = "";
        Font_Char_Set = 0;
        Font_Size = 0.0;
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "Universe.Snapshot";
        Control_On = Yes;
        Control_On_Int = 1;
        Control_On_Long = 0;
        Control_On_Double = 1.0;
        Control_On_String = "";
        Control_Off = No;
        Control_Off_Int = 0;
        Control_Off_Long = 0;
        Control_Off_Double = 0.0;
        Control_Off_String = "";
        Toggle_Button_On = No;
        Shortcut_Key = "";
    }
    Panel_Item {
        Item_Type = Slider;
        Item_Name = "Num of Rulers";
        Display_Area = 0, 0, 0, 0;
        Font_Name = "";
        Font_Char_Set = 0;
        Font_Size = 0.0;
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "UNIVERSE.NumOfRulers";
        Slider_Range_Max = 998;
        Slider_Value_Min = 2.0;
```

```
    Slider_Value_Step = 1.0;
    Slider_Value = "98";
  }
  Panel_Item {
    Item_Type = Slider;
    Item_Name = "Num of Periods";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "Universe.TimeManager";
    Slider_Range_Max = 999;
    Slider_Value_Min = 1.0;
    Slider_Value_Step = 1.0;
    Slider_Value = "499";
  }
  Panel_Item {
    Item_Type = Slider;
    Item_Name = "Relationship";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "Universe.Relationship";
    Slider_Range_Max = 2;
    Slider_Value_Min = 0.0;
    Slider_Value_Step = 1.0;
    Slider_Value = "0";
  }
  Panel_Item {
    Item_Type = Slider;
    Item_Name = "Govt. Mobilization";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "Universe.RulerControl(0)";
    Slider_Range_Max = 400;
    Slider_Value_Min = 1.0;
    Slider_Value_Step = 0.01;
    Slider_Value = "0";
  }
  Panel_Item {
    Item_Type = Slider;
    Item_Name = "ExInput to Govt.";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
```

```
    Var_Name = "Universe.RulerControl(1)";
    Slider_Range_Max = 1000;
    Slider_Value_Min = 0.0;
    Slider_Value_Step = 1.0;
    Slider_Value = "0";
  }
  Panel_Item {
    Item_Type = Slider;
    Item_Name = "PC Resources";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "UNIVERSE.RPC";
    Slider_Range_Max = 1000;
    Slider_Value_Min = 0.0;
    Slider_Value_Step = 1.0;
    Slider_Value = "0";
  }
  Panel_Item {
    Item_Type = Slider;
    Item_Name = "SInsgt. Mobilization";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "Universe.RulerControl(5)";
    Slider_Range_Max = 400;
    Slider_Value_Min = 1.0;
    Slider_Value_Step = 0.01;
    Slider_Value = "0";
  }
  Panel_Item {
    Item_Type = Slider;
    Item_Name = "ExInput to SInsgt.";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Var_Name = "Universe.RulerControl(6)";
    Slider_Range_Max = 1000;
    Slider_Value_Min = 0.0;
    Slider_Value_Step = 1.0;
    Slider_Value = "0";
  }
  Panel_Item {
    Item_Type = Slider;
    Item_Name = "Mobilization Limit";
    Display_Area = 0, 0, 0, 0;
    Font_Name = "";
    Font_Char_Set = 0;
    Font_Size = 0.0;
```

```
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "Universe.RulerControl(2)";
        Slider_Range_Max = 400;
        Slider_Value_Min = 1.0;
        Slider_Value_Step = 0.01;
        Slider_Value = "0";
    }
    Panel_Item {
        Item_Type = Slider;
        Item_Name = "Coercion Effect";
        Display_Area = 0, 0, 0, 0;
        Font_Name = "";
        Font_Char_Set = 0;
        Font_Size = 0.0;
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "Universe.PotentialControl(0)";
        Slider_Range_Max = 100;
        Slider_Value_Min = 0.0;
        Slider_Value_Step = 0.01;
        Slider_Value = "20";
    }
    Panel_Item {
        Item_Type = Slider;
        Item_Name = "Mobilization Effect";
        Display_Area = 0, 0, 0, 0;
        Font_Name = "";
        Font_Char_Set = 0;
        Font_Size = 0.0;
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "Universe.PotentialControl(1)";
        Slider_Range_Max = 300;
        Slider_Value_Min = 0.0;
        Slider_Value_Step = 0.01;
        Slider_Value = "0";
    }
    Panel_Item {
        Item_Type = Slider;
        Item_Name = "Weight to Frontline";
        Display_Area = 0, 0, 0, 0;
        Font_Name = "";
        Font_Char_Set = 0;
        Font_Size = 0.0;
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "Universe.RulerControl(4)";
        Slider_Range_Max = 100;
        Slider_Value_Min = 0.0;
        Slider_Value_Step = 0.1;
        Slider_Value = "20";
    }
    Panel_Item {
        Item_Type = Slider;
        Item_Name = "Threat Threshold";
        Display_Area = 0, 0, 0, 0;
```

```
        Font_Name = "";
        Font_Char_Set = 0;
        Font_Size = 0.0;
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "Universe.RelationalControl(0)";
        Slider_Range_Max = 100;
        Slider_Value_Min = 0.0;
        Slider_Value_Step = 0.1;
        Slider_Value = "10";
    }
    Panel_Item {
        Item_Type = Slider;
        Item_Name = "Decentralization";
        Display_Area = 0, 0, 0, 0;
        Font_Name = "";
        Font_Char_Set = 0;
        Font_Size = 0.0;
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "Universe.RelationalControl(1)";
        Slider_Range_Max = 100;
        Slider_Value_Min = 0.0;
        Slider_Value_Step = 0.01;
        Slider_Value = "0";
    }
    Panel_Item {
        Item_Type = Slider;
        Item_Name = "Noise Level";
        Display_Area = 0, 0, 0, 0;
        Font_Name = "";
        Font_Char_Set = 0;
        Font_Size = 0.0;
        Font_Style = 0;
        Font_Color = 0, 0, 0;
        Font_Background_Color = 0, 0, 0;
        Var_Name = "Universe.Noise";
        Slider_Range_Max = 199;
        Slider_Value_Min = 0.01;
        Slider_Value_Step = 0.01;
        Slider_Value = "19";
    }
}


//-------------------------------------------------------
// Space Initial Value Definition Divsion
//-------------------------------------------------------
Space_Initial {
}


//-------------------------------------------------------
// Other Miscellaneous Data Definition Division
//-------------------------------------------------------
Others_Data {
    Indent = 4;
    Rule_Colored = Yes;
    Font_Name = "Dialog";
```

```
    Font_Char_Set = 0;
    Font_Size = 12.0;
    Font_Style = 0;
    Font_Color = 0, 0, 0;
    Font_Background_Color = 0, 0, 0;
    Window_Hide_Run = 1;
    Debug_Position_x = 75;
    Debug_Position_y = 75;
    Debug_Size_x = 400;
    Debug_Size_y = 200;
    Console_Position_x = 1047;
    Console_Position_y = 5;
    Console_Size_x = 388;
    Console_Size_y = 578;
    Componen_Tree_Position_x = 2;
    Componen_Tree_Position_y = 0;
    Componen_Tree_Size_x = 218;
    Componen_Tree_Size_y = 449;
}


//-------------------------------------------------------
// Network Information Definition Division
//-------------------------------------------------------
Network_Info {
    Global_Space_Area = 0, 0, 100, 100;
    Local_Space_Info {
    }
}
```