Grammar - British

Description and Dependencies

Contents

[Introduction 5](#_Toc58763111)

[Problem Statement 5](#_Toc58763112)

[Content 5](#_Toc58763113)

[Hierarchies and grammar 6](#_Toc58763114)

[Hierarchy Definition 6](#_Toc58763115)

[Shield 6](#_Toc58763116)

[Grammar 6](#_Toc58763117)

[Examples 6](#_Toc58763118)

[Simplest Shield 6](#_Toc58763119)

[Field 6](#_Toc58763120)

[Grammar 7](#_Toc58763121)

[Division 7](#_Toc58763122)

[Division by 2 8](#_Toc58763123)

[Grammar 8](#_Toc58763124)

[Examples 8](#_Toc58763125)

[Simple Division 2 Fields 8](#_Toc58763126)

[Simple Division Shield 9](#_Toc58763127)

[Positioned Halves 9](#_Toc58763128)

[Line Variation 9](#_Toc58763129)

[Division by 3 9](#_Toc58763130)

[Grammar 9](#_Toc58763131)

[Examples 9](#_Toc58763132)

[Division by 4 9](#_Toc58763133)

[Complex Divisions 9](#_Toc58763134)

[Tincture 10](#_Toc58763135)

[Grammar 10](#_Toc58763136)

[Examples 11](#_Toc58763137)

[Simple Tincture 11](#_Toc58763138)

[Reference 11](#_Toc58763139)

[Proper 11](#_Toc58763140)

[Simple Fur 11](#_Toc58763141)

[Vair 11](#_Toc58763142)

[Vaire 11](#_Toc58763143)

[Counter 11](#_Toc58763144)

[Vaire Between 11](#_Toc58763145)

[Field Variation 12](#_Toc58763146)

[Grammar 12](#_Toc58763147)

[Examples 13](#_Toc58763148)

[Field Variation 2 tinctures 13](#_Toc58763149)

[Orientation 13](#_Toc58763150)

[Semy 13](#_Toc58763151)

[Semy Of Charge 13](#_Toc58763152)

[Known Semy 13](#_Toc58763153)

[Charge 14](#_Toc58763154)

[Grammar 14](#_Toc58763155)

[Simple Charge 15](#_Toc58763156)

[Grammar 15](#_Toc58763157)

[Examples 15](#_Toc58763158)

[Single Simple Charge 15](#_Toc58763159)

[Multi Simple Charge 15](#_Toc58763160)

[Shared Properties After Tincture 15](#_Toc58763161)

[Field Variation 16](#_Toc58763162)

[Shared Properties Before Tincture 16](#_Toc58763163)

[Multi Charges 17](#_Toc58763164)

[Grammar 17](#_Toc58763165)

[Examples 17](#_Toc58763166)

[List with And Separator 17](#_Toc58763167)

[List With Coma Separator 17](#_Toc58763168)

[List With Positioned Charges 17](#_Toc58763169)

[List With Coma and Positioned Charges 18](#_Toc58763170)

[List containing a On Charge 18](#_Toc58763171)

[List Containing a Charged Charge 18](#_Toc58763172)

[List containing a Surmounted Charge 18](#_Toc58763173)

[List Within a charge 18](#_Toc58763174)

[Charge Between Position 19](#_Toc58763175)

[Grammar 19](#_Toc58763176)

[Examples 19](#_Toc58763177)

[Between Start 19](#_Toc58763178)

[Between Middle 19](#_Toc58763179)

[Charge On Position 21](#_Toc58763180)

[Grammar 21](#_Toc58763181)

[Examples 21](#_Toc58763182)

[On Start 21](#_Toc58763183)

[On Middle 21](#_Toc58763184)

[Charge Within 22](#_Toc58763185)

[Grammar 22](#_Toc58763186)

[Examples 22](#_Toc58763187)

[Simple Within 22](#_Toc58763188)

[Single Positioned Charge Within 22](#_Toc58763189)

[Positioned Charges Within 22](#_Toc58763190)

[Charge Surmounted 23](#_Toc58763191)

[Grammar 23](#_Toc58763192)

[Examples 23](#_Toc58763193)

[Simple Surmounted 23](#_Toc58763194)

[Multiple Surmounted with Positions 24](#_Toc58763195)

[Charge Overall 25](#_Toc58763196)

[Grammar 25](#_Toc58763197)

[Examples 25](#_Toc58763198)

[Simple Overall 25](#_Toc58763199)

[Overall On Charge 25](#_Toc58763200)

[Overall Charge Charged 25](#_Toc58763201)

[Charge Charged 27](#_Toc58763202)

[Grammar 27](#_Toc58763203)

[Examples 27](#_Toc58763204)

[Charged Simple 27](#_Toc58763205)

[Charged Positioned 27](#_Toc58763206)

[Charged Charges List 27](#_Toc58763207)

[Charged Surmounted 27](#_Toc58763208)

[Symbol 27](#_Toc58763209)

[Symbol Cross 29](#_Toc58763210)

[Ordinary 29](#_Toc58763211)

[Grammar 29](#_Toc58763212)

[Examples 29](#_Toc58763213)

[Alteration 29](#_Toc58763214)

[Simple Ordinary 30](#_Toc58763215)

[Line Variation 30](#_Toc58763216)

[Location 31](#_Toc58763217)

# Introduction

## Problem Statement

The grammar is the set of rules that enable to read and understand a blazon string in order to turn it into a proper format. The grammar is very close to the language it supports, the blazon may use a limited set of terms, but the construction is highly tied to the language complexity it stems from (English here).

Establishing an exhaustive set of grammar rules to match all the example possible is the target.

## Content

In this document the grammar will be presented in the basic Backus–Naur form (I am not a specialist so I might not properly use this form). With explanation of the topics and why they are split in those specific grammar rules. The grammar is highly inspired by the rules of blazon, and try to cover all the cases in the most straightforward way (might not always be achievable given the amount of exceptions)

Every grammar rule will be expressed against the grammar hierarchy. For the lower level of the hierarchy examples of blazon that require those grammar will be provided as well as the reasoning behind the rules / exceptions.

# Hierarchies and grammar

The goal is to establish the ER Diagram representing all the relations between all the different grammar rules. Identify the different loops. Given the complexity of the grammar the graph will be split in family blocks, sized by complexity.

### Hierarchy Definition

The hierarchy graph is a representation of the link that the different grammar have with each other (which grammar needs which other grammar). The hierarchies are kept small based on readability and concepts, no specific rule for this.

The colour code is as follow:

**Blue** for the grammars of interest in the current section

**Green** for the grammars that have been presented somewhere else

**Black** for the grammar that are not yet solved / available

## Shield

Second step would be to lay out the whole grammar as an exhaustive Backus–Naur document

The shield hierarchy is relatively simple, the field is the basic of the coat of arms, followed optionally by the charge(s), optional attribute for all counterchanged or sign of cadency are presented at that level as well.

### Grammar

Shield := Field. | Field. LightSeparator! (Charge)+ AllCounterChanged? Cadency?

### Examples

#### Simplest Shield

Simplest is subjective, but to only contain one word a shield can be defined as just one field, with just one simple tincture

Example: Argent

## Field

The field is the first and only required element of the shield.

The commencement of any blazon is of necessity a description of the field, the one word signifying its colour being employed if it be a simple field; or, if it be composite, such terms as are necessary.

The field family is composed of the following objects

### Grammar

The field can only be composed either of a division, a tincture or a field variation

Field := Division | Tincture | FieldVariation

## Division

The division are another way of filling the shield, it does not necessarily end up creating multiple “sub shield” and can still be considered as one field, or it can be creating multiple “sub shield” with every part of the division having its own blazon. It is mostly prevalent in quarterly cases. Division can also be used in dividing charges, even animals.

A division can be divided again, so that a division by 2, in turn divided by 3 become a division by 6. But, the objects necessary for the division are just 2 and 3.

Notes that the division of the field does NOT guarantee that the following field content will be respecting the division. If only one charge is described after the division, where should this charge be located (is it an implied overall?).

The grammar will be presented by division types

## Division by 2

The division family is big, containing the simplest 2 division, the simple 3 and 4 or the more complex and rarer customized divisions (like 3 and 2 or other exotic partitions). The complexity comes from the different ways to express a division by 2. Why do we need the 3 grammatical rules (simple division by 2 field, simple division shield, positioned halves)?

### Grammar

Division := DivisionBy2 | DivisionBy3 | DivisionBy4 | ComplexDivision

DivisionBy2 := DivisionBy2Name. LightSeparator? LineVariationDefinition? (SimpleDivisionBy2Field | SimpleDivisionShield | PositionedHalves)

DivisionBy2Name := "Party Per Fess" | "Party Per Pale" | "Party Per Bend" | "Party Per Bend Sinister" | …

LightSeparator := ","

LineVariationDefinition := LineVariation. (Counter. LineVariation.)?

LineVariation := "Indented" | "Dancetty" | "Dentilly" | "Rayonny" | …

Counter := "Counter"

SimpleDivisionBy2Field := (Tincture | FieldVariation) And. (Tincture | FieldVariation)

And := "And"

SimpleDivisionShield := Shield. LightSeparator? And. Shield.

PositionedHalves := FirstDivisionNumber. Shield. ChargeSeparator? SecondDivisionNumber. Shield.

FirstDivisionNumber := "1"

SecondDivisionNumber := "2"

### Examples

#### Simple Division 2 Fields

The division name is per saltire, there are no light separator, no line variation and then a simple division by 2 with ermine and azure as tinctures

Example: Per Bend ermine and azure

#### Simple Division Shield

Example: TBD

#### Positioned Halves

Example: Per fess, 1 per pale gules an escutcheon argent an escarbuncle Or and Or a fess chequy argent and gules, 2 azure three fleurs-de-lys or a bordure compony argent and gules.

Example: Parted Per Bend Sinister, 1 Ermine, 2 Azure

#### Line Variation

Example: Parted per pale indented Sable and Argent, two harts’ attires counterchanged and on a chief Gules a crescent Or between two Ermine spots

Example: Parted per pale Bretesse Ermine and Azure

## Division by 3

### Grammar

DivisionBy3 := DivisionBy3Name. LightSeparator? LineVariationDefinition? SimpleDivisionBy3Field

DivisionBy3Name := “Tierced Per Fess” | “Tierced Per Pale” | …

SimpleDivisionBy3Field := (Tincture | FieldVariation) LightSeparator. (Tincture | FieldVariation) And (Tincture | FieldVariation)

### Examples

#### Simple Tierced

Example: Tierced per pale, gules, argent and azure

Example: per Pall Gules, azure and Argent

## Division by 4

### Grammar

DivisionBy4 := DivisionBy4Name. DivisionBy4Separator? LineVariationDefinition? (SimpleDivisionBy2Field | SimpleDivisionShield | PositionnedQuarters)

DivisionBy4Name := “Quarterly” | “Party Per Cross” | “Per Saltire” | …

DivisionBy4Separator := “,” | “;” | “:”

SimpleDivisionShield := Shield. LightSeparator? And. Shield.

PositionnedQuarters := (FirstDivisionNumber. Quarter? Shield. ChargeSeparator. SecondDivisionNumber. Quarter? Shield. ChargeSeparator. ThirdDivisionNumber. Quarter? Shield. ChargeSeparator. FourthDivisionNumber. Quarter? Shield. Separator?)

| FirstAndFourthDivisionNumber. Quarter? Shield. ChargeSeparator. SecondAndThirdDivisionNumber. Quarter? Shield. Separator?)

| FirstAndFourthDivisionNumber. Quarter? Shield. ChargeSeparator. SecondDivisionNumber. Quarter? Shield. ChargeSeparator. ThirdDivisionNumber. Quarter? Shield. Separator?)

| FirstDivisionNumber. Quarter? Shield. ChargeSeparator. SecondAndThirdDivisionNumber. Quarter? Shield. ChargeSeparator. FourthDivisionNumber. Quarter? Shield. Separator?)

TBD

### Examples

TBD

## Complex Divisions

## Tincture

The tincture is a critical part of the grammar and part of the minimum path for a valid blazon “Argent” would be my favourite minimum and is a tincture metal. Tincture are applied on fields and charges, and can be complex (for vairé definitions)

### Grammar

Tincture := (SimpleTincture | TinctureFur | Counterchanged)

SimpleTincture := (TinctureColour | TinctureMetal | Tincture Reference | Tincture Proper)

TinctureColour := "Azure" | "Gules" | …

TinctureMetal := "Argent" | "Or"

TinctureReference := "Of the last" | " Of the Field" | "Of the second" | …

Tincture Proper := "Proper”

TinctureFur := (SimpleFur | Vair | Vaire)

SimpleFur := "Ermines" | "Pean" | …

Vair := Counter ? VairName. (SymbolStateDeterminer? FurOrientationName)?

VairName := "Vair" | "Potent" | …

SymbolStateDeterminer := TBD

FurOrientationName := TBD

Vaire := Counter? VaireName (SymbolStateDeterminer? FurOrientationName)? SimpleTincture And SimpleTincture | SimpleTincture VaireBetweenName SimpleTincture

VaireName := "Vaire" | "Potente" | …

VaireBetweenName := "Papelonne" | "Masoned"

### Examples

#### Simple Tincture

Example: Quarterly Gules and Or

#### Reference

Example: Argent, on a Bend Sable, three Owls of the First.

#### Proper

Example: Gules, a bend argent between two fountains proper.

#### Simple Fur

Example: Azure, a pale engrailed Ermines

#### Vair

Example: Gules, a fess Vair

#### Vaire

Example: Vairé or and gules

#### Counter

Example: Counter Potenté Argent and sable

#### Vaire Between

Example: gules masoned argent

## Field Variation

The field variation is a big family usually changing the field (but can also be applied to a charge, because why use a term like field, if not to violate its usage the next second?).

The grammar complexity is so advanced in the charge definition, that the case when a semy is using a charge to paint the field, the charge described is forced to be a simpler subset version of the Simple Charge. The semy charge is only used here and is not a valid “charge” definition outside of the semy usage.

### Grammar

FieldVariation := FieldVariation2Tinctures | FieldVariationSemy | FieldVariationKnownSemy

FieldVariation2Tinctures := FieldVariationName. Orientation? (Of. FieldVariationNumber)? Tincture. And. Tincture. | Tincture. FieldVariationName. Orientation? Tincture.

FieldVariationName := "Barruly" | "BarryPily" | "Barry" | …

Orientation := TBD

Of := "Of"

FieldVariationNumber := "Two" | "Three" | "Four" | "Five" | …

FieldVariationSemy := SimpleTincture ((SemyDeterminer? Semy. SemyCharge) | SemyName) SimpleTincture?

SemyCharge := ChargeElement. SimpleTincture.

SemyDeterminer := "A" | "An"

Semy := "Semy" | "Semy Of"

SemyName := "Crusily" | "Billety" | "Annulletty" | …

FieldVariationKnownSemy := SimpleTincture. PredefinedSemy.

PredefinedSemy := "Bezanty" | "Hurty" | "Platy" | …

### Examples

#### Field Variation 2 tinctures

Example: barruly of twelve argent and gules

#### Orientation

TBD

#### Semy

Example: Gules Billety Or

#### Semy Of Charge

Example: Or a semy of hearts Gules

#### Known Semy

Example: Gules platy

## Charge

The most complex and differentiated content of any blazon the charges are very complex to describe as a grammar.

The charge can be positioned, can be predefined (the name implies the properties) can have orientation, can be relative to other charges, can be numbered, can be used as a variation of the field, etc …

The charge is divided in the most common cases, in increasing complexity.

Semy are the charges that are used to fill a field through semy, the particularity of those charges is that they cannot be clearly positioned (position is always implied by the field variation). They do not have a number, no orientation (I think), and cannot be filled with a field variation of their own (again I think)

Simple Charge is the grammar that represent the happy path of charges in a blazon, a number, a name a tincture or field variation, those charges are the main subject and not used as a filling and thus are meant to have shared properties (location, position, orientation …). They are split in between single and plural since some grammar rules (presented below) does require a specific number of charges. But both of those share the same definition, the only difference being the determiner pointing toward a 1 for single.

### Grammar

Charge := SimpleCharge | MultiCharges | TBD

SimpleCharge := SingleSimpleCharge | PluralSimpleCharge

## Simple Charge

The simple charge is the most basic of all the possible charges, can be used a standalone in a valid coat of arms. This contains a determiner, the charge element, potential properties and a filling, either as a field variation or as a tincture. The hierarchy used is the plural simple charge, as it is a subset of the single simple charge, the single simple charge has the same grammar just with a “1” determiner limitation.

### Grammar

SingleSimpleCharge := SingleDeterminer? ChargeElement. ((Tincture | FieldVariation)? SharedProperties | SharedProperties? (Tincture | FieldVariation))

PluralSimpleCharge := PluralDeterminer? ChargeElement. ((Tincture | FieldVariation)? SharedProperties | SharedProperties? (Tincture | FieldVariation))

SingleDeterminer := “A” | “An” | “The” | “One”

PluralDeterminer := “Two” | “2” | “Three” | …

ChargeElement := Ordinary | Symbol | SymbolCross

SharedProperties := SharedKeyword? (SharedObjectReference | SharedPropertyAdverb)\* SharedProperty+

SharedKeyword := “All” | “Each”

SharedObjectReference := “Of the last” | “Of the first” | “Pair” | …

SharedPropertyAdverb := “Devouring” | “Charged” | “With” | …

SharedProperty := Direction | Location

Direction := “Paleways” | “Fessways” | “Crosswise” | …

### Examples

#### Single Simple Charge

Example: ARGENT, A LION PASSANT GULES

#### Multi Simple Charge

Example: OR, THREE HAMADES GULES

#### Shared Properties After Tincture

TBD

#### Field Variation

Example: Azure a lion rampant barry of ten argent and gules, armed or

#### Shared Properties Before Tincture

Example: Ermine, a sword paleways Proper

## Multi Charges

Multi charges represent a more complex representation of charges grammar when multiple charges are present on the field. There is multiple variation, and they are not all incompatible, the subject is complex enough to have its own documentation.

As presented below the Charge list (A and B and C is an example) could be composed of other multi charges, it is true for most of them and the compatibility rule is documented in the charge group documentation

### Grammar

MultiCharges := ChargesList | ChargeBetweenPosition | ChargeOnPosition | ChargeWithinPosition | ChargeSurmounted | ChargeOverall | ChargeCharged

ChargesList := ANDPOSSIBLEGROUP (ANDListSEPARATOR ANDPOSSIBLEGROUP)\*

AndPossibleGroup := SimpleCharge | ChargeOnPosition | ChargeBetweenPosition | ChargeCharged

AndListSeparator := “And” | “,”

ChargeBetweenPosition := BetweenMiddle | BetweenStart

ChargeOnPosition := OnStart | OnMiddle

ChargeSurmounted := SurmountedSingle | SurmountedPlural

ChargeOverall := OverallPossibleFirstGroup OverallKeyword OverallPossibleSecondGroup

ChargeCharged := SimpleCharge ChargedKeyword ChargePossibleGroup

### Examples

#### List with And Separator

Example: ARGENT, AN OAK TREE VERT AND A CHIEF SABLE

#### List With Coma Separator

Example: TBF

#### List With Positioned Charges

Example: Azure, two eagles displayed in chief and a mullet in base argent

#### List With Coma and Positioned Charges

Example: Gules, a castle surmounted with a tower argent; in base a lion passant gardant or

#### List containing a On Charge

Example: Vert, a fess chequy Argent and Azure between three cuirasses (or habergeons) Argent and on a chief Argent three buckles Azure

#### List Containing a Charged Charge

Example: Sable, a naked man Proper and a dexter canton Argent charged with a sword and pistol in saltire Gules

#### List containing a Surmounted Charge

Example: Gules, a boar passant Or and a canton Ermine charged with a sword paleways Proper surmounted by a crescent for difference

#### List Within a charge

Example: Argent, a chevron between two cinquefoils in chief Gules and a saltire couped Azure in base all within a bordure Gules

## Charge Between Position

### Grammar

ChargeBetweenPosition := BetweenMIDDLE | BetweenSTART

BetweenMIDDLE := BetweenPOSSIBLEFIRSTGROUP (ChargesList)\* BetweenKEYWORD BetweenPOSSIBLESECONDGROUP (BetweenKEYWORD BetweenPOSSIBLESECONDGROUP)\*

BetweenSTART := BetweenKEYWORD BetweenPOSSIBLEFIRSTGROUP BetweenPOSSIBLESECONDGROUP (BetweenKEYWORD BetweenPOSSIBLESECONDGROUP)\*

BetweenKeyword := “Between”

BetweenPOSSIBLEFIRSTGROUP := (SIMPLECHARGE | ChargesList | SURMOUNTED | CHARGED)

BetweenPOSSIBLESECONDGROUP := (PLURALSIMPLECHARGE | ChargesList | PLURALSURMOUNTED | PLURALCHARGED)

### Examples

#### Between Start

Example: Argent, between two chevrons sable three ashen keys vert

#### Between Middle

Example: Or, a cross between four keys gules

#### Between On Charges

Example: Gules, on a bend argent between two fountains proper, a rose gules between two mullets sable.

## Charge On Position

### Grammar

ON := ONSTART | ONMIDDLE

ONSTART := ONKEYWORD ONPOSSIBLEGROUP ONPOSSIBLEGROUP

ONMIDDLE := (SIMPLECHARGE | DIVISIONOFTHEFIELD) ONEACHKEYWORD ONPOSSIBLEGROUP

ONPOSSIBLEGROUP := (SIMPLECHARGE | BETWEENCOMMON | SURMOUNTED | CHARGED)

### Examples

#### On Start

Example: Argent, on a chief Gules three pallets Or

#### On Middle

Example: Gules, five fusils in fess argent on each an escallop sable

## Charge Within

### Grammar

ChargeWithin := WITHINPOSSIBLEFIRSTGROUP (ChargesList | WHOLEKEYWORD)? WITHINKEYWORD WITHINPOSSIBLESECONDGROUP

| WITHINPOSSIBLEFIRSTGROUP WITHINKEYWORD WITHINPOSSIBLESECONDGROUP ALL WITHINKEYWORD WITHINPOSSIBLESECONDGROUP

| WITHINPOSSIBLEFIRSTGROUP ALL WITHINKEYWORD WITHINPOSSIBLESECONDGROUP WHOLEKEYWORD WITHINPOSSIBLESECONDGROUP

WITHINPOSSIBLEFIRSTGROUP := SIMPLECHARGE | ChargesList | ChargeBETWEEN | ChargeSurmounted | ChargeCHARGED

WITHINPOSSIBLESECONDGROUP := SIMPLECHARGE |ChargesList | ChargeCHARGED

### Examples

#### Simple Within

Example: Argent, two chevrons within a bordure engrailed Gules

#### Single Positioned Charge Within

Example: Azure, (3, 3, 2, 1) nine stars Argent within a bordure wavy Or

#### Positioned Charges Within

Example: Party per fesse or and sable, in chief a greyhound courant in base an owl within a bordure engrailed all counter-changed

## Charge Surmounted

Obviously, the concept of surmounting a field is a grammar “container” all the leaves used in this grammar are OTHER charge grammars.

### Grammar

SURMOUNTEDPOSSIBLEFIRSTSINGLEGROUP := SINGLESIMPLECHARGE | SINGLECHARGED

SURMOUNTEDPOSSIBLEFIRSTPLURALGROUP := PLURALSIMPLECHARGE | PLURALCHARGED

SURMOUNTEDPOSSIBLESECONDGROUP := SIMPLECHARGE | BETWEENMIDDLE | CHARGED

SURMOUNTEDSINGLE := SURMOUNTEDPOSSIBLEFIRSTSINGLEGROUP SUMOUNTEDKEYWORD SURMOUNTEDPOSSIBLESECONDGROUP

SURMOUNTEDPLURAL := SURMOUNTEDPOSSIBLEFIRSTPLURALGROUP EACHKEYWORD SUMOUNTEDKEYWORD SURMOUNTEDPOSSIBLESECONDGROUP

ChargeSurmounted := SURMOUNTEDSINGLE | SURMOUNTEDPLURAL

### Examples

#### Simple Surmounted

Example: Argent, three piles Sable surmounted by a fess wavy Gule

#### Multiple Surmounted with Positions

Example: Azure, three stars Argent and in the centre a cross Argent surmounted by a saltire Gules and in dexter chief a crescent surmounted by a mullet for difference

## Charge Overall

### Grammar

OVERALLPOSSIBLEFIRSTGROUP := SIMPLECHARGE | ON | AND | CHARGED | SURMOUNTED | BETWEEN | WITHIN | DIVISIONOFTHEFIELD

OVERALLPOSSIBLESECONDGROUP := SIMPLECHARGE | ChargeOn | ChargeCharged

OVERALL := OVERALLPOSSIBLEFIRSTGROUP OVERALLKEYWORD OVERALLPOSSIBLESECONDGROUP

### Examples

#### Simple Overall

Example: Quarterly Or and Gules and overall a bend Sable.

#### Overall On Charge

Example: Azure, two wings conjoined argent over all on a fess gules three bezants.

#### Overall Charge Charged

Example: Paly of six Argent and Gules and overall a bend Azure charged with three cushions Argent

## Charge Charged

### Grammar

CHARGEDPOSSIBLEGROUP := SIMPLECHARGE | AND | BETWEENMIDDLE | SURMOUNTED

ChargeCharged := SIMPLECHARGE CHARGEDKEYWORD CHARGEDPOSSIBLEGROUP

### Examples

#### Charged Simple

Example: Or, three bars wavy Gules each charged with an escallop Or

#### Charged Positioned

Example: Argent, a saltire and chief Gules, the last charged with a mullet Or in dexter chief all within a bordure indented Gules

#### Charged Charges List

Example: Argent, an orle Gules and in chief three martlets Sable all within a bordure Azure charged with thistles, roses, fleurs-de-lis and harps alternately all Or

#### Charged Surmounted

Example: Gules, a boar passant Or and a canton Ermine charged with a sword paleways Proper surmounted by a crescent for difference

## Symbol

## Symbol Cross

## Ordinary

Ordinary are part of the simple object that can be placed as a charge. They have rules and can virtually be describe exhaustively. Unlike Symbols that can be infinite in their description the Ordinary are well defined.

### Grammar

Ordinary := OrdernaryAlteration? SimpleOrdinary | MultiOrdinary

OrdinaryAlteration := “Half” |”Quarter” | “Double”

SimpleOrdinary := (OrdinaryHonourable | OrdinarySubOrdinary | OrdinaryDiminutive) LineVariationDefinition?

OrdinaryHonourable := “Cross” | “Pale” | “Fess” | …

OrdinarySubordinary := OrdinaryMobile | OrdinaryFixed

OrdinaryMobile := “Escutcheon” | “Lozenge” | “Fusil” | …

OrdinaryFixed := “Canton” | “Flaunches” | “Fret” | …

OrdinaryDiminutive := “Label” | “Pallet” | “Endorse” | …

LineVariationDefinition := “Indented” | “Dancetty” | “Dentilly” | …

### Examples

#### Alteration

Example: Or, a chevron Sable between three martlets Gules within a double tressure flory counter-flory also Gules

#### Simple Ordinary

Example: Argent, a pale Sable

#### Line Variation

Example: Vert, a fess dancetty Ermine

## Location