Grammar - British

Description and Dependencies

Contents

[Introduction 7](#_Toc60501176)

[Problem Statement 7](#_Toc60501177)

[Content 7](#_Toc60501178)

[Hierarchies and grammar 8](#_Toc60501179)

[Hierarchy Definition 8](#_Toc60501180)

[Shield 8](#_Toc60501181)

[Grammar 8](#_Toc60501182)

[Examples 8](#_Toc60501183)

[Simplest Shield 8](#_Toc60501184)

[Field 8](#_Toc60501185)

[Grammar 8](#_Toc60501186)

[Division 9](#_Toc60501187)

[Division by 2 10](#_Toc60501188)

[Grammar 10](#_Toc60501189)

[Examples 10](#_Toc60501190)

[Simple Division 2 Fields 10](#_Toc60501191)

[Simple Division Shield 11](#_Toc60501192)

[Simple Division As Single Field 11](#_Toc60501193)

[Positioned Halves 11](#_Toc60501194)

[Line Variation 11](#_Toc60501195)

[Division by 3 12](#_Toc60501196)

[Grammar 12](#_Toc60501197)

[Examples 12](#_Toc60501198)

[Simple Tierced 12](#_Toc60501199)

[Division by 4 13](#_Toc60501200)

[Grammar 14](#_Toc60501201)

[Examples 14](#_Toc60501202)

[Complex Divisions 13](#_Toc60501203)

[Tincture 13](#_Toc60501204)

[Grammar 14](#_Toc60501205)

[Examples 14](#_Toc60501206)

[Simple Tincture 14](#_Toc60501207)

[Reference 14](#_Toc60501208)

[Proper 14](#_Toc60501209)

[Simple Fur 14](#_Toc60501210)

[Vair 14](#_Toc60501211)

[Vaire 14](#_Toc60501212)

[Counter 14](#_Toc60501213)

[Vaire Between 14](#_Toc60501214)

[Field Variation 13](#_Toc60501215)

[Grammar 14](#_Toc60501216)

[Examples 14](#_Toc60501217)

[Field Variation 2 tinctures 14](#_Toc60501218)

[Orientation 14](#_Toc60501219)

[Semy 14](#_Toc60501220)

[Semy Of Charge 14](#_Toc60501221)

[Known Semy 14](#_Toc60501222)

[Semy followed by a Charge (Semy Charge detection) 14](#_Toc60501223)

[Charge 13](#_Toc60501224)

[Type of charges 13](#_Toc60501225)

[Semy Charges 13](#_Toc60501226)

[Simple Charge 13](#_Toc60501227)

[Multi Charges 13](#_Toc60501228)

[Grammar 14](#_Toc60501229)

[Single Simple Charge 15](#_Toc60501230)

[Grammar 15](#_Toc60501231)

[Examples 15](#_Toc60501232)

[Single Simple Charge 15](#_Toc60501233)

[Single Simple Ordinary 15](#_Toc60501234)

[Single Simple Ordinary The 15](#_Toc60501235)

[Plural Simple Charge 16](#_Toc60501236)

[Grammar 16](#_Toc60501237)

[Examples 16](#_Toc60501238)

[Multi Simple Charge 16](#_Toc60501239)

[Shared Properties After Tincture 16](#_Toc60501240)

[Multi Plural Ordinary 16](#_Toc60501241)

[Field Variation 17](#_Toc60501242)

[Shared Properties Before Tincture 17](#_Toc60501243)

[Simple Charge with no Determiner 17](#_Toc60501244)

[Multi Charges 18](#_Toc60501245)

[Grammar 18](#_Toc60501246)

[Examples 18](#_Toc60501247)

[List with And Separator 19](#_Toc60501248)

[List With Coma Separator 19](#_Toc60501249)

[List With Positioned Charges 19](#_Toc60501250)

[List With Coma and Positioned Charges 19](#_Toc60501251)

[List containing a On Charge 19](#_Toc60501252)

[List Containing a Charged Charge 19](#_Toc60501253)

[List containing a Surmounted Charge 19](#_Toc60501254)

[List Within a charge 19](#_Toc60501255)

[List without Separator But location 19](#_Toc60501256)

[Charge Between Position 20](#_Toc60501257)

[Grammar 21](#_Toc60501258)

[Examples 21](#_Toc60501259)

[Between Start 21](#_Toc60501260)

[Between Middle 21](#_Toc60501261)

[Between On Charges 21](#_Toc60501262)

[Charge On Position 20](#_Toc60501263)

[Grammar 20](#_Toc60501264)

[Examples 20](#_Toc60501265)

[On Start 20](#_Toc60501266)

[On Middle 20](#_Toc60501267)

[Charge Within 21](#_Toc60501268)

[Grammar 22](#_Toc60501269)

[Examples 22](#_Toc60501270)

[Simple Within 22](#_Toc60501271)

[Single Positioned Charge Within 22](#_Toc60501272)

[Positioned Charges Within 22](#_Toc60501273)

[Charge Surmounted 21](#_Toc60501274)

[Grammar 21](#_Toc60501275)

[Examples 21](#_Toc60501276)

[Simple Surmounted 21](#_Toc60501277)

[Multiple Surmounted with Positions 22](#_Toc60501278)

[Charge Overall 23](#_Toc60501279)

[Grammar 24](#_Toc60501280)

[Examples 24](#_Toc60501281)

[Simple Overall 24](#_Toc60501282)

[Overall On Charge 24](#_Toc60501283)

[Overall Charge Charged 24](#_Toc60501284)

[Charge Charged 23](#_Toc60501285)

[Grammar 24](#_Toc60501286)

[Examples 24](#_Toc60501287)

[Charged Simple 24](#_Toc60501288)

[Charged Positioned 24](#_Toc60501289)

[Charged Charges List 24](#_Toc60501290)

[Charged Surmounted 24](#_Toc60501291)

[Symbol 23](#_Toc60501292)

[Grammar 24](#_Toc60501293)

[Examples 24](#_Toc60501294)

[Simple Symbol 24](#_Toc60501295)

[Symbol Alteration 24](#_Toc60501296)

[Symbol Attitude 24](#_Toc60501297)

[Symbol Attitude with Attribute 24](#_Toc60501298)

[Symbol With subparts 24](#_Toc60501299)

[Symbol with All the Simple grammar 24](#_Toc60501300)

[Symbol Cross 23](#_Toc60501301)

[Ordinary 24](#_Toc60501302)

[Grammar 24](#_Toc60501303)

[Single Ordinary 25](#_Toc60501304)

[Grammar 25](#_Toc60501305)

[Examples 25](#_Toc60501306)

[Chief 25](#_Toc60501307)

[Quarter 25](#_Toc60501308)

[Canton 25](#_Toc60501309)

[Inescutcheon 25](#_Toc60501310)

[Bordure 25](#_Toc60501311)

[Fret 25](#_Toc60501312)

[Label 25](#_Toc60501313)

[Line Variation 25](#_Toc60501314)

[Single Ordinary with Plural Ordinary Name 26](#_Toc60501315)

[Plural Ordinary 27](#_Toc60501316)

[Grammar 27](#_Toc60501317)

[Examples 27](#_Toc60501318)

[Cross 27](#_Toc60501319)

[Pale 27](#_Toc60501320)

[Fess 27](#_Toc60501321)

[Bar 27](#_Toc60501322)

[Bend & Bend Sinister 27](#_Toc60501323)

[Chevron 27](#_Toc60501324)

[Saltire 27](#_Toc60501325)

[Pile 27](#_Toc60501326)

[Flaunches 27](#_Toc60501327)

[Orle 28](#_Toc60501328)

[Pall 28](#_Toc60501329)

[Location 29](#_Toc60501330)

[Annexes 30](#_Toc60501331)

[Full Grammar 30](#_Toc60501332)

[Successful Parsing Examples 30](#_Toc60501333)

[Known Limitations 30](#_Toc60501334)

# 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 number 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 has 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 either a tincture, a field variation (with multiple objects on it) or a division. The division of the field, can be then used as a unique field, for the charges, or be terminal, in the sense that all the part of the division are their own shield as well.

### 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

#### Simple Division As Single Field

Here the division is not used as the foundation for one shield per divided area, but as a description of the field, that is as unique a background as if it were a single tincture

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

#### Semy followed by a Charge (Semy Charge detection)

This is meant to test that the charge detection can separate the charge of a semy (hearts gules) which is simple, and not include a real charge on top of it (a lion rampant azure) which does belong to the field. Since charges are based off unknown character the determiner separates if the charge is a real charge or a property of the charge (lion rampant is not a property of hearts gules)

Example: Or a semy of hearts Gules a lion rampant Azure

## 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 notion of “charge” is very generic, and regroup the simplest single honourable like a chief, as well as complex, relatively positioned list of charges with potential advanced symbols

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

### Type of charges

All the charges presented here are not part of the current description of the particular “charge” grammar (this grammar implies a charge on a field). But they are considered charges nonetheless and will have their own grammar to avoid incompatibility with other grammar while parsing.

#### Semy Charges

Semy charges 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

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 and grammatical construction, the only difference being the single or multiple determiners.

In the context of this grammar a number followed by a charge name is not considered a list of charges, even though there are multiple instances of symbol that are represented by this “simple charge”.

#### Multi Charges

Multi charges are representing collection of charges that have a relationship between each other, and can share multiple properties, including tincture. The multi charge grammar is complex and extensively use the notion of “simple charge” for each unit within the collection.

### Grammar

Charge := SimpleCharge | MultiCharges | TBD

SimpleCharge := SingleSimpleCharge | PluralSimpleCharge

## Single Simple Charge

### Grammar

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

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

SingleChargeElement := SingleOrdinary | 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

#### Single Simple Ordinary

Example: Chequy or and azure, a quarter ermine

#### Single Simple Ordinary The

Example: Azure, the moon Argent

## Plural 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. The reason for splitting the charges this way is to separate the types of charges in the complex scenario when some grammar of complex charges can only be applied if one charge of the other is singular or plural.

Until I find an example, I will assume that a simple charge, always starts with a determiner. This is an easy (not enough though) way to separate the charge subpart with a real charge

### Grammar

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

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

PluralChargeElement := PluralOrdinary | 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

#### Multi Simple Charge

Example: Or, Three Hamades Gules

#### Shared Properties After Tincture

TBD

#### Multi Plural Ordinary

Example: Or, three pallets Gules

#### Field Variation

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

The problem with this example is the fact that the “simple charge” parent grammar is trying to include the field variation or tincture as part of its first-generation definition. But a simple charge is too generic, the behaviour difference between an honourable and a symbol is too wide to be refactored at the simple charge level. In this scenario the symbol sub part “armed or” is after the “simple charge” field variation, but of course a simple charge does NOT have a grammar support for a symbol sub part. Thus, the field variation and tincture should be part of the charge element level (even its children). Now since we know we can have a shared property before the tincture, we also need to ship the shared properties at the level of the charge element child. For more complex scenario where multiple charges are being listed with common properties, this will be covered in the complex charges, and we need to make sure the simple ones are still compatible children of complex charges.

#### Shared Properties Before Tincture

Example: Ermine, a sword paleways Proper

#### Simple Charge with no Determiner

Example: TBF

## Multi Charges

Multi charges represent a more complex set 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

The only way to have more than 1 symbol in a list is for the symbol to have a location. It is possible to have multiple ordinary without stated location because some ordinary have default location. So pretty much the grammar for a list of multi charge should be located simple charge, not simple charge. A located simple charge can be a symbol with a location on the field, or an ordinary with the same.

So pretty much we can have:

Symbol + Ordinary

Ordinary + Ordinary

Symbol + Located Symbol

Located Symbol + Symbol

Located Symbol + Located Symbol

### Grammar

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

ChargesList := AndPossibleGroup ((Coma AndPossibleGroup)\* And AndPossibleGroup).

AndPossibleGroup := SimpleCharge | ChargeOnPosition | ChargeBetweenPosition | ChargeCharged

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: Azure, (3, 3, 2, 1) nine stars Argent, a crescent for difference

#### 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

Example: Argent, a double-headed eagle displayed Sable and on a chief Vert two mullets Or

#### 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

#### List without Separator But location

Example: Party per fesse or and sable, in chief a greyhound proper courant in base an owl of the first

## Charge Between Position

### Grammar

ChargeBetweenPosition := BetweenMiddle | BetweenStart

BetweenMiddle := BetweenPossibleFirstGroup Between BetweenPossibleSecondGroup (Between BetweenPossibleSecondGroup)\*

BetweenStart := Between BetweenPossibleSecondGroup BetweenPossibleFirstGroup

Between := “Between”

BetweenInsideGroup := (SimpleCharge | ChargesList | Surmounted | Charged)

BetweenSurroundingGroup := (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 Start With Charges List

Example: TBD

#### Between Middle With Charges List

Example: TBD

#### Between On Charges

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

#### Multiple Between

Example: TBD

## Charge On Position

### Grammar

ChargeOnPosition := OnStart | OnMiddle

OnStart := On OnPossibleGroup OnPossibleGroup

OnMiddle := (SimpleCharge | Division) OnEach OnPossibleGroup

OnPossibleGroup := (SimpleCharge | ChargeBetweenPosition | SURMOUNTED | CHARGED)

On := “On”

OnEach := “On Each”

### 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 := WithinWhole | WithinAll | WithinAllWhole

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

WithinAll := WITHINPOSSIBLEFIRSTGROUP WITHINKEYWORD WITHINPOSSIBLESECONDGROUP ALL WITHINKEYWORD WITHINPOSSIBLESECONDGROUP

WithinAllWhole := 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

### Grammar

Symbol := SymbolAlteration? SymbolName. SymbolAttitude\* SymbolAttitudeAttribute? SharedProperty? (Tincture | FieldVariation) SymbolSubpartList\*

SymbolSubpartList := SymbolSubpartGroup. LightSeparator. (SymbolSubpartGroup\* And SymbolSubpartGroup)?

SymbolSubpartGroup := SubpartNameList. SimpleTincture.

SymbolSubPartNameList := SubpartName. ((LightSeparator. SubpartName)\* And SubpartName.)\*

### Examples

#### Simple Symbol

Example: or, a lion gules

#### Symbol Alteration

Example: Azure, a double-headed eagle displayed Or

#### Symbol Attitude

Example: Argent, a lion salient Gules

#### Symbol Attitude with Attribute

Example: Gules, a lion rampant regardant or

#### Symbol With subparts

Example: argent a lion gules armed and langued or

In the following example the simple tincture is included INSIDE of the subpart group

Example: Argent a lion rampant Purpure crowned Or, langued and armed Gules;

#### Symbol with All the Simple grammar

Example: argent a demi-lion rampant issuant sable armed and langued gules

## Symbol Cross

## Symbol With Implied Tincture

Crescent seems to have an implied texture but can also be specified, the other ones are simplifications of charge with a tincture that can’t be replaced.

## 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.

I start wondering if I should not consider one grammar rule per ordinary (or at least for the most specials, like bordure, orle, point …)

### Grammar

Ordinary := OrdernaryAlteration? SingleOrdinary | PluralOrdinary

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

PluralOrdinary := Cross | OrdinaryMobile | OrdinaryDiminutive

## Single Ordinary

Single ordinary is ordinary that can be described as a single element, virtually all the ordinary, even the diminution can be set alone.

Since all ordinary that can be defined as plural can also be defined alone, we can extend the single ordinary with the plural ordinaries.

List of ordinaries that are always singular in a blazon:

Chief, Quarter, Canton, Inescutcheon, Bordure, Fret, Label, Gyron

### Grammar

SingleOrdinary := (SingleOrdinaryName | PluralOrdinaryName) LineVariationDefinition?

SingleOrdinaryName := “Chief” | “Quarter” | “Canton” | “Inescutcheon” | “Bordure” | “Fret” | “Label”

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

### Examples

#### Chief

Example: Argent masonry sable, a chief indented of the second.

#### Quarter

Example: Chequy or and azure, a quarter ermine

#### Canton

Example: Sable, four bars Argent; the canton ermine

#### Inescutcheon

Example: Argent a cross Gules and thereon an inescutcheon Azure, three fleurs-de-lis Or.

#### Bordure

Example: argent, two bars azure within a bordure engrailed sable.

Example: Gules three crescents argent within a bordure of the last charged with eight roses of the first.

#### Fret

Example: argent a fret sable

#### Label

Example: Gules, three lions passant guardant Or, with a label of three points Argent each point charged with three torteaux Gules

#### Line Variation

Example: or, a cross engrailed within a bordure invecked, sable

Example: Or, on a chief embattled Sable three escallops Or

#### Single Ordinary with Plural Ordinary Name

Example: Argent, a pale Sable

## Plural 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. So far I did not find any ordinary that can only be plural and never be blazoned in a singular form.

### Grammar

PluralOrdinary := PluralOrdinaryName LineVariationDefinition?

PluralOrdinaryName := “Cross” | “Pallet” | “Fess” | …

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

### Examples

#### Cross

Example: Azure, three crosses crosslet fitch Argent

#### Pale

Example: Argent, Two pales Sable

#### Fess

I could not find an example of a blazon that does have multiple fesses described on the field

Example: Chequy argent and sable, a fesse gules

#### Bar

Example: Or, three bars gules

#### Bend & Bend Sinister

Example: Gules, three bends or a chief per fess ermine and argent

#### Chevron

Example: or three chevrons gules

#### Saltire

Example: Gules, three cinquefoils Argent within a bordure Argent charged with four saltires couped and as many mullets alternately Gules

#### Pile

Example: Argent, three piles Sable

#### Flaunches

Example: Sable, a lion rampant between two flaunches or

#### Orle

Example: Argent, a lion rampant gules an orle of nine pheons azure

#### Pall

Example: Argent, three pallets azure, a bend sinister wavy Or.

## Location

# Annexes

## Full Grammar

## Successful Parsing Examples

## Known Limitations