



Grammar

Maturity

Model

Dr. Vadim Zaytsev, Universiteit van Amsterdam @ GEMOC'14

@grammarware

Grammars in a Broad Sense

- Definitions of languages
 - Finite definitions
 - of infinite languages
- Focus
 - SE: sets of programs (analytic sense, “parsing”)
 - FL: sets of words (generative sense)
- Also, rewriting systems

Arithmetic Expressions, Boolean Expressions, and Expressions

$$\begin{aligned} \langle \text{factor} \rangle &::= \langle \text{number} \rangle \bar{\text{or}} \langle \text{function} \rangle \bar{\text{or}} \langle \text{variable} \rangle \bar{\text{or}} \langle \text{subscr var} \rangle \\ &\quad \bar{\text{or}} (\langle \text{ar exp} \rangle) \bar{\text{or}} \langle \text{factor} \rangle \uparrow \langle \text{ar exp} \rangle \downarrow \\ \langle \text{term} \rangle &::= \langle \text{factor} \rangle \bar{\text{or}} \langle \text{term} \rangle \times \langle \text{factor} \rangle \bar{\text{or}} \langle \text{term} \rangle / \langle \text{factor} \rangle \\ \langle \text{ar exp} \rangle &::= \langle \text{term} \rangle \bar{\text{or}} + \langle \text{term} \rangle \bar{\text{or}} - \langle \text{term} \rangle \bar{\text{or}} \langle \text{ar exp} \rangle + \langle \text{term} \rangle \\ &\quad \bar{\text{or}} \langle \text{ar exp} \rangle - \langle \text{term} \rangle \\ \langle \text{ar exp A} \rangle &::= \langle \text{ar exp} \rangle \\ \langle \text{relation} \rangle &::= \langle \bar{\text{or}} \rangle \bar{\text{or}} \leq \bar{\text{or}} \geq \bar{\text{or}} = \bar{\text{or}} \neq \\ \langle \text{rel exp} \rangle &::= (\langle \text{ar exp} \rangle \langle \text{relation} \rangle \langle \text{ar exp A} \rangle) \\ \langle \text{bool term} \rangle &::= 0 \bar{\text{or}} 1 \bar{\text{or}} \langle \text{rel exp} \rangle \bar{\text{or}} \langle \text{function} \rangle \bar{\text{or}} \\ &\quad \langle \text{variable} \rangle \bar{\text{or}} \langle \text{subscr var} \rangle \bar{\text{or}} (\langle \text{bool exp} \rangle) \\ &\quad \bar{\text{or}} \neg \langle \text{bool term} \rangle \\ \langle \text{bool exp} \rangle &::= \langle \text{bool term} \rangle \bar{\text{or}} \langle \text{bool exp} \rangle \vee \langle \text{bool term} \rangle \\ &\quad \bar{\text{or}} \langle \text{bool exp} \rangle \wedge \langle \text{bool term} \rangle \bar{\text{or}} \\ &\quad \langle \text{bool exp} \rangle \equiv \langle \text{bool term} \rangle \\ \langle \text{exp} \rangle &::= \langle \text{ar exp} \rangle \bar{\text{or}} \langle \text{bool exp} \rangle \end{aligned}$$

$$\begin{array}{lcl}
S & \longrightarrow & aSB \& AB \mid b \\
A & \longrightarrow & aA \mid \varepsilon \\
B & \longrightarrow & B_1 \mid B_2 \\
B_1 & \longrightarrow & B_1B_3 \& B_2B_2 \mid b \\
B_2 & \longrightarrow & B_1B_1 \& B_2B_6 \mid bb \\
B_3 & \longrightarrow & B_1B_2 \& B_6B_6 \mid bbb \\
B_6 & \longrightarrow & B_1B_2 \& B_3B_3
\end{array}$$

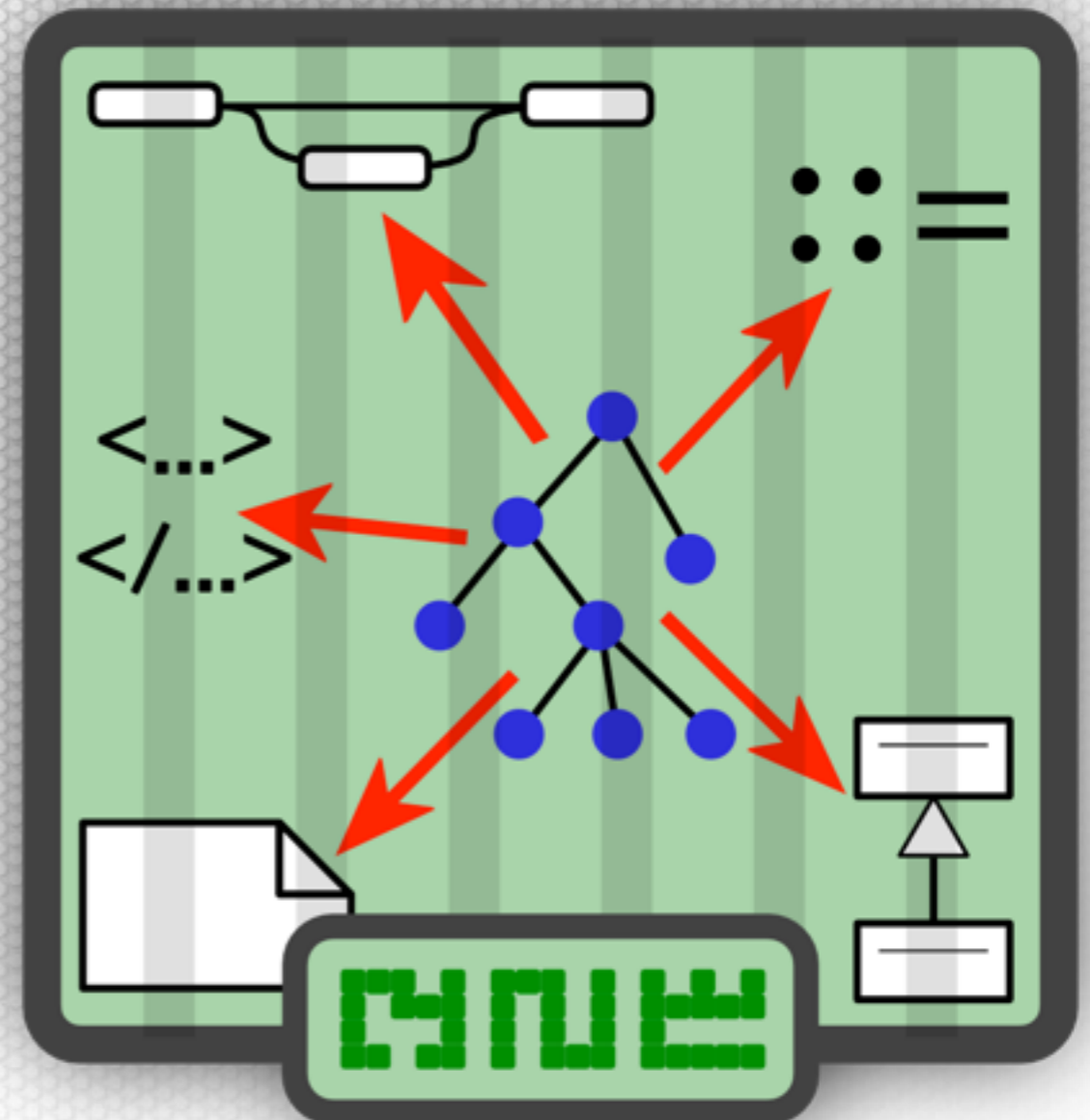
Grammars in a Broad Sense

- Grammars as
 - contracts
 - protocols
 - schemata
 - domain models
 - format definitions
 - ...

metamodels

Grammar Zoo

- Collect ALL the grammars
 - Atlantic, ANTLR, TXL, OMG, ASF+SDF, RELAX, W3C, ...



ISO/IEC 1989:2014 - Inform...

www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=51416

Français | Русский | Member area

ISO

Standards | About us | Standards Development | News | Store

Standards catalogue | Online collections | Graphical symbols

Search ISO

Store > Standards catalogue > By ICS > JTC 1 Information technology > SC 22

ISO/IEC 1989:2014

Information technology -- Programming languages, their environments and system software interfaces -- Programming language COBOL

Abstract [Preview ISO/IEC 1989:2014](#)

ISO/IEC 1989:2014 specifies the syntax and semantics of COBOL. Its purpose is to promote a high degree of machine independence to permit the use of COBOL on a variety of data processing systems.

ISO/IEC 1989:2014 specifies:

- the form of a compilation group written in COBOL;
- the effect of compiling a compilation group;
- the effect of executing run units;
- the elements of the language for which a conforming implementation is required to supply a definition;
- the elements of the language for which meaning is explicitly undefined;
- the elements of the language that are dependent on the capabilities of the processor.

ISO/IEC 1989:2014 does not specify:

- the means whereby a compilation group written in COBOL is compiled into code executable by a processor;
- the time at which method, function, or program runtime modules are linked or bound to an activating statement, except that runtime binding occurs of necessity when the identification of the appropriate program or method is not known at compile time;
- the time at which parameterized classes and interfaces are expanded;
- the mechanism by which locales are defined and made available on a processor;

FORMAT ?

LANGUAGE

PDF English

PDF English

CHF 198 [Add to basket](#)

Keep up to date with ISO
Sign up to our newsletter for the latest news, views and product information

[Subscribe](#)

Contact customer services
[Send your enquiry by email](#)
or call us on +41 22 749 08 88

Lecture - Free Photos For C++ x ISO/IEC JTC1/SC22/WG21 - x Software Language Process: x Grammar Zoo x http://www.omg.org/spec/...

www.open-std.org/JTC1/SC22/WG21/

Welcome to the official home of



JTC1/SC22/WG21 - The

2014-07-10: [standards](#) | [projects](#) | [papers](#) | [mailings](#) | [internals](#) | [m](#)

News 2014-07-09: The deadline for the next mailing is 2014-10-1
News 2014-07-09: The [2014-07-post-Rapperswil mailing](#) is availa
News 2014-07-09: The [C++ Standard Core Language Issues List](#) (Revision 90) is available ([.zip](#))
News 2014-05-29: The [C++ Standard Library Issues List](#) (Revision 88) is available ([.zip](#))
News 2013-05-24: The [CD](#) for the new C++ standard is released
News 2013-05-24: New [ub](#) reflector and archive. for undefined behaviour study group.
News 2013-01-06: New [ranges](#) and [features](#) reflectors and archives.
News 2011-09-11: The new C++ standard - C++11 - is published!

[ISO/IEC JTC1/SC22/WG21](#) is the international standardization working group for the programming language C++.

Published [standards and technical reports](#) include:

- [ISO/IEC 14882:2011 Programming Language C++ - draft](#)
- [ISO/IEC TR 18015:2006 C++ Performance - draft TR](#)

Work on [projects](#) and their [milestones](#) include:

- [ISO/IEC 14882: Programming Language C++ - latest publicly available draft](#)
- [ISO/IEC TR 24733: C++ decimal floating point arithmetic extensions - draft](#)
- [ISO/IEC 29124: C++ Special Math Functions - draft](#)

Other information:

- [Some further information on C++ standardization - isocpp.org](#)
- [C++ Standard Core Issues List](#)
- [C++ Standard Library Issues List](#)
- [WG papers](#)
- [WG mailings](#)
- [WG21 business plan and convener's report 2011](#)
- [Information on past and future WG meetings](#)
- [Contacts: membership, liaisons and related work](#)
- [ISO/IEC TR 24772 Information technology -- Programming languages -- Guidance to avoiding vulnerabilities in programming languages through language selection and use](#)
- [WG internal information](#) (For members of the C++ standardization committee)

If you want further information, or want to participate, please contact your national [member body](#) or one of the [contact addresses](#) of the WG.

2014-07-10: [standards](#) | [projects](#) | [papers](#) | [mailings](#) | [internals](#) | [meetings](#) | [contacts](#)

Authentication Required

The server http://www.open-std.org:80 requires a username and password. The server says: SC22/WG21 C++.

User Name:

Password:

Cancel Log In

NOT
fetched

zaytsev — ghc — 66x18

Last login: Sun Sep 28 09:19:54 on ttys003

>>>>14:27<<<<< ~ zaytsev\$ ghci

GHCi, version 7.6.3: <http://www.haskell.org/ghc/> :? for help

Loading package ghc-prim ... linking ... done.

Loading package integer-gmp ... linking ... done.

Loading package base ... linking ... done.

Prelude>

NOT

fetches

NOT
quite

fetches

AWK(1)

awk

NAME

awk - pattern-directed scanning and processing language

SYNOPSIS

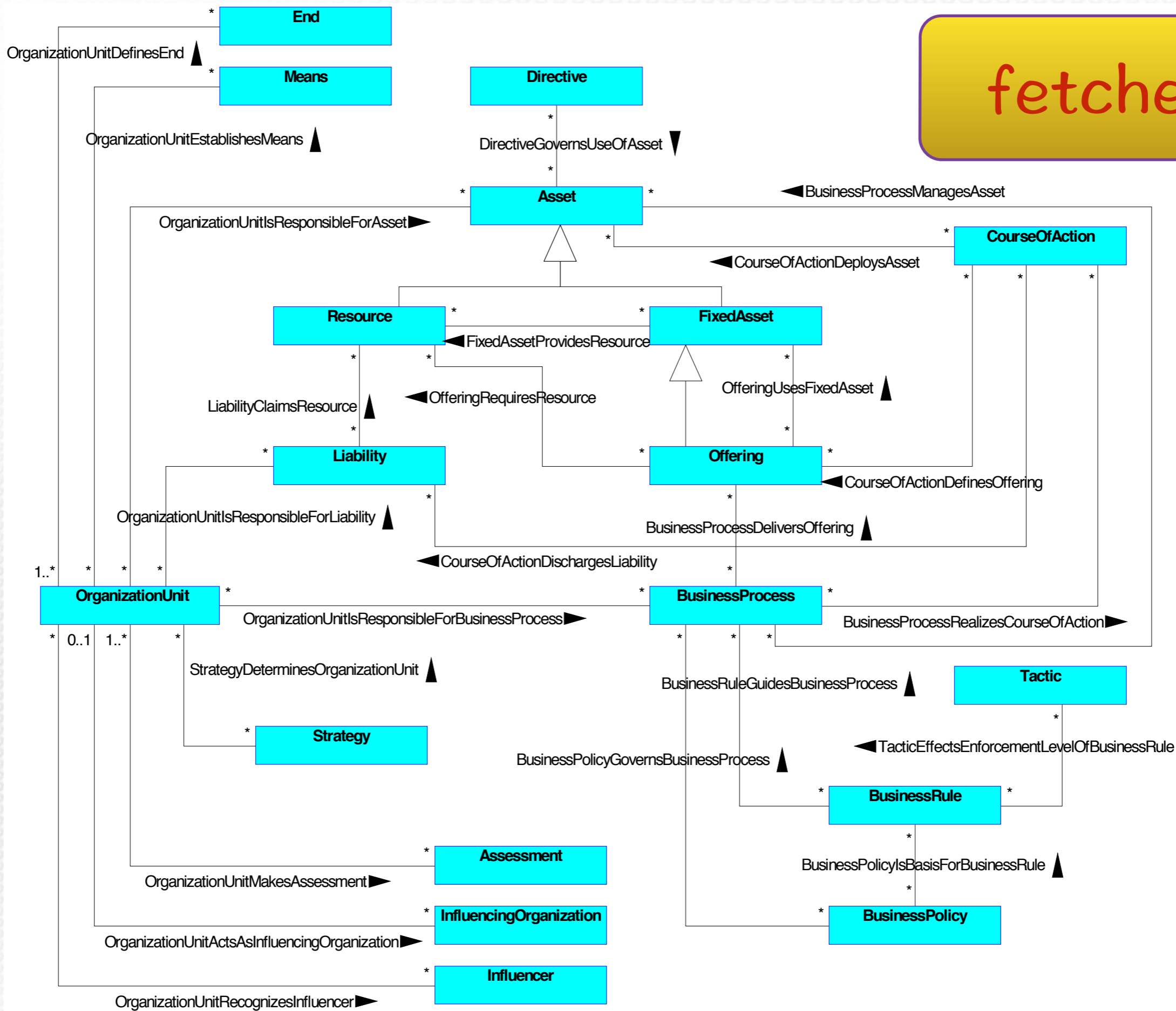
awk [-F fs] [-v var=value] ['prog' | -f progfile] [file ...]

DESCRIPTION

Awk scans each input file for lines that match any of a set of patterns specified literally in prog or in one or more files specified as -f progfile. With each pattern there can be an associated action that will be performed when a

: █

AWK(1)




fetches

```
<?xml version="1.0" encoding="UTF-8"?>
<uml:Model xmi:version = "2.1" xmlns:xmi = "http://schema.omg.org/spec/XMI/2.1" xmlns:uml = "http://www.eclipse.org/uml2/2.1.0/UML" name = "BMM"
xmi:id = "_cda4594e-69ce-419b-b16d-798a9cb4d9a4package">
  <packagedElement xmi:type = "uml:Package" xmi:id = "_cda4594e-69ce-419b-b16d-798a9cb4d9a4" name = "BMM" visibility = "public">
    <packagedElement xmi:type = "uml:Class" xmi:id = "_6151e0df-f072-4cf5-9fdd-7ae1569b0bd5" name = "Assessment" isAbstract = "FALSE" visibility =
"public">
      <generalization xmi:id = "_e5854fb5-8f7f-4e8b-b162-2853044bca98" general = "_e73a4961-91c7-4ff4-8787-209c85faf718">
        </generalization>
        <ownedAttribute xmi:id="_476c8095-c44b-4b03-89a0-fc34b02e55cc" name = "categorizingAssessmentCategory" type = "_86d0a586-a581-4570-86c6-
7f295f44f1a0" default = "" visibility = "private" aggregation = "none" association = "" redefinedProperty = "" isReadOnly = "FALSE" isUnique = "FALSE"
isStatic = "false">
          <upperValue xmi:type = "uml:LiteralUnlimitedNatural" xmi:id = "_476c8095-c44b-4b03-89a0-fc34b02e55ccupper" value = "-1"/>
          <lowerValue xmi:type = "uml:LiteralInteger" xmi:id = "_476c8095-c44b-4b03-89a0-fc34b02e55cclower"/>
        </ownedAttribute>
        <ownedAttribute xmi:id="_4825b10a-bd59-44f8-b4ea-40839908cd35" name = "affectedMeans" type = "_90d0feea-3bca-4f73-b408-a9c603b6692a" default =
"" visibility = "private" aggregation = "none" association = "" redefinedProperty = "" isReadOnly = "FALSE" isUnique = "FALSE" isStatic = "false">
          <upperValue xmi:type = "uml:LiteralUnlimitedNatural" xmi:id = "_4825b10a-bd59-44f8-b4ea-40839908cd35upper" value = "-1"/>
          <lowerValue xmi:type = "uml:LiteralInteger" xmi:id = "_4825b10a-bd59-44f8-b4ea-40839908cd35lower"/>
        </ownedAttribute>
        <ownedAttribute xmi:id="_a33022cd-7cd6-4d45-8d8d-7f2e3878c6a0" name = "affectedEnd" type = "_25259062-fdf9-4978-b028-c6a0f61baeed" default = ""
visibility = "private" aggregation = "none" association = "" redefinedProperty = "" isReadOnly = "FALSE" isUnique = "FALSE" isStatic = "false">
          <upperValue xmi:type = "uml:LiteralUnlimitedNatural" xmi:id = "_a33022cd-7cd6-4d45-8d8d-7f2e3878c6a0upper" value = "-1"/>
          <lowerValue xmi:type = "uml:LiteralInteger" xmi:id = "_a33022cd-7cd6-4d45-8d8d-7f2e3878c6a0lower"/>
        </ownedAttribute>
        <ownedAttribute xmi:id="_eaae3981-12ec-4e64-b44f-d8cdb72cfd2f" name = "usedAssessment" type = "_6151e0df-f072-4cf5-9fdd-7ae1569b0bd5" default =
"" visibility = "private" aggregation = "none" association = "" redefinedProperty = "" isReadOnly = "FALSE" isUnique = "FALSE" isStatic = "false">
          <upperValue xmi:type = "uml:LiteralUnlimitedNatural" xmi:id = "_eaae3981-12ec-4e64-b44f-d8cdb72cfd2fupper" value = "-1"/>
          <lowerValue xmi:type = "uml:LiteralInteger" xmi:id = "_eaae3981-12ec-4e64-b44f-d8cdb72cfd2flower"/>
        </ownedAttribute>
        <ownedAttribute xmi:id="_4f1de0df-6dd3-4360-912c-f55904368c80" name = "usingAssessment" type = "_6151e0df-f072-4cf5-9fdd-7ae1569b0bd5" default =
"" visibility = "private" aggregation = "none" association = "" redefinedProperty = "" isReadOnly = "FALSE" isUnique = "FALSE" isStatic = "false">
          <upperValue xmi:type = "uml:LiteralUnlimitedNatural" xmi:id = "_4f1de0df-6dd3-4360-912c-f55904368c80upper" value = "-1"/>
          <lowerValue xmi:type = "uml:LiteralInteger" xmi:id = "_4f1de0df-6dd3-4360-912c-f55904368c80lower"/>
        </ownedAttribute>
        <ownedAttribute xmi:id="_15d22738-69fe-4db5-b002-05068df7a680" name = "motivatedDirective" type = "_8dc5d579-d6aa-4053-ac1d-8f50cc0cfb7f"
default = "" visibility = "private" aggregation = "none" association = "" redefinedProperty = "" isReadOnly = "FALSE" isUnique = "FALSE" isStatic =
"false">
          <upperValue xmi:type = "uml:LiteralUnlimitedNatural" xmi:id = "_15d22738-69fe-4db5-b002-05068df7a680upper" value = "-1"/>
          <lowerValue xmi:type = "uml:LiteralInteger" xmi:id = "_15d22738-69fe-4db5-b002-05068df7a680lower"/>
        </ownedAttribute>
        <ownedAttribute xmi:id="_9642458f-4ab9-4cbc-b577-5b0283b2dfd5" name = "identifiedPotentialImpact" type = "_ed5248f5-58b9-442e-8b90-f009c5d80fb7"
default = "" visibility = "private" aggregation = "none" association = "" redefinedProperty = "" isReadOnly = "FALSE" isUnique = "FALSE" isStatic =
"false">
          <upperValue xmi:type = "uml:LiteralUnlimitedNatural" xmi:id = "_9642458f-4ab9-4cbc-b577-5b0283b2dfd5upper" value = "-1"/>
          <lowerValue xmi:type = "uml:LiteralInteger" xmi:id = "_9642458f-4ab9-4cbc-b577-5b0283b2dfd5lower"/>
        </ownedAttribute>
        <ownedAttribute xmi:id="_c484c833-786b-4f2d-890c-39e066368cc5" name = "judgedInfluencer" type = "_263eea9e-e465-48df-bac6-af14775c4f47" default
= "" visibility = "private" aggregation = "none" association = "" redefinedProperty = "" isReadOnly = "FALSE" isUnique = "FALSE" isStatic = "false">
          <upperValue xmi:type = "uml:LiteralUnlimitedNatural" xmi:id = "_c484c833-786b-4f2d-890c-39e066368cc5upper" value = "-1"/>
          <lowerValue xmi:type = "uml:LiteralInteger" xmi:id = "_c484c833-786b-4f2d-890c-39e066368cc5lower" value = "1"/>
        </ownedAttribute>
        <ownedAttribute xmi:id="_0f49bd6c-5856-4d6d-957a-b4f0cfa200b3" name = "assessingOrganizationUnit" type = "_ddb18c38-08fe-4b79-83db-ea72c3ff32f1"
default = "" visibility = "private" aggregation = "none" association = "" redefinedProperty = "" isReadOnly = "FALSE" isUnique = "FALSE" isStatic =
"false">
          <upperValue xmi:type = "uml:LiteralUnlimitedNatural" xmi:id = "_0f49bd6c-5856-4d6d-957a-b4f0cfa200b3upper" value = "-1"/>
          <lowerValue xmi:type = "uml:LiteralInteger" xmi:id = "_0f49bd6c-5856-4d6d-957a-b4f0cfa200b3lower" value = "1"/>
        </ownedAttribute>
      </packagedElement>
      <packagedElement xmi:type = "uml:Class" xmi:id = "_86d0a586-a581-4570-86c6-7f295f44f1a0" name = "AssessmentCa
visibility = "public">
        <generalization xmi:id = "_6ffb03f3-e036-424b-8646-584ee6c7da04" general = "_e73a4961-91c7-4ff4-8787-209c85
        </generalization>
        <ownedAttribute xmi:id=" 68a00296-9b7a-456b-b561-ce3efd159006" name = "categorizedAssessment" type = " 6151
```

fetches

file:///Users/zaytsev/projects/zoo/web/dsl/business/motivation/v1.0/atlanctic1/extracted/index.html

Browsable Business Motivation Model (BMM) Grammar



Grammar extracted by [Vadim Zaytsev](#), see the [Grammar Zoo](#) entry for details: [dsl/business/motivation/v1.0/atlanctic1/extracted](#)
 Source used for this grammar: OMG, *The Business Motivation Model from the OMG*, written from the OMG BMM specification, generated from the CMOF XMI for BMM 1.0, bmmOmg.ecore, January 2010

Summary

- Total **62** production rules with **62** top alternatives and **480** symbols.
- Vocabulary: **122** = **32** nonterminals + **0** terminals + **90** labels + **0** markers.
- Total **32** nonterminal symbols: **32** defined (see below), **0** root (—), **2** top ([MotivationElement](#), [String](#)), **0** bottom (—).

Syntax

```
Assessment ::=
    [motivatedDirective]::Directive+ [affectedEnd]::End+ [judgedInfluencer]::Influencer+ [assessingOrganizationUnit]::OrganizationUnit
```

```
MotivationElement ::=
    Assessment
```

```
MotivationElement ::=
    Means
```

```
MotivationElement ::=
    OrganizationUnit
```

```
MotivationElement ::=
    End
```

```
MotivationElement ::=
    BusinessProcess
```

```
MotivationElement ::=
    Asset
```

```
MotivationElement ::=
    Liability
```

```
MotivationElement ::=
    Influencer
```

Browsable Business Motivation Model (BMM) Grammar



Grammar extracted by [Vadim Zaytsev](#), see the [Grammar Zoo](#) entry for details: [dsl/business/motivation/v1.0/atlantic1/extracted](#)
Source used for this grammar: OMG, *The Business Motivation Model from the OMG*, written from the OMG BMM specification, generated from the CMOF XMI for BMM 1.0, bmmOmg.ecore, January 2010

Summary

- Total **62** production rules with **62** top alternatives and **480** symbols.
- Vocabulary: **122** = **32** nonterminals + **0** terminals + **90** labels + **0** markers.
- Total **32** nonterminal symbols: **32** defined (see below), **0** root (—), **2** top (**MotivationElement**, **String**), **0** bottom (—).
- Total **90** labels: `[motivatedDirective]`⁴, `[affectedEnd]`, `[judgedInfluencer]`, `[assessingOrganizationUnit]`, `[affectedMeans]`, `[identifiedPotentialImpact]`, `[usedAssessment]`, `[usingAssessment]`, `[categorizingAssessmentCategory]`, `[supportedDesiredResult]`², `[governedCourseOfAction]`, `[derivedCourseOfAction]`, `[governedAsset]`, `[motivatingPotentialImpact]`, `[directiveRegulation]`, `[motivatingAssessment]`, `[establishingOrganizationUnit]`, `[judgingAssessment]`³, `[definedEnd]`, `[managedBusinessProcess]`, `[managedLiability]`, `[managedAsset]`², `[determiningStrategy]`, `[recognizedInfluencer]`, `[internalInfluencingOrganization]`, `[madeAssessment]`, `[establishedMeans]`, `[definingOrganizationUnit]`, `[deliveredOffering]`, `[realizedCourseOfAction]`, `[governingBusinessPolicy]`, `[guidingBusinessRule]`, `[responsibleOrganizationUnit]`³, `[definingCourseOfAction]`, `[usedAsset]`, `[requiredResource]`, `[deliveringBusinessProcess]`, `[deployingCourseOfAction]`, `[governingDirective]`², `[managingBusinessProcess]`, `[enabledCourseOfAction]`, `[enablingCourseOfAction]`, `[moreSpecificCourseOfAction]`, `[broaderCourseOfAction]`, `[definedOffering]`, `[dischargedLiability]`, `[realizingBusinessProcess]`, `[baseDirective]`, `[deployedAsset]`, `[moreSpecificDesiredResult]`, `[broaderDesiredResult]`, `[supportingDirective]`, `[supportingCourseOfAction]`, `[claimedResource]`, `[dischargingCourseOfAction]`, `[providingFixedAsset]`, `[requiringOffering]`, `[claimingLiability]`, `[usingOffering]`, `[providedResource]`, `[moreSpecificBusinessPolicy]`, `[broaderBusinessPolicy]`, `[derivedBusinessRule]`, `[governedBusinessProcess]`, `[effectingTactic]`, `[guidedBusinessProcess]`, `[baseBusinessPolicy]`, `[implementedStrategy]`, `[enforcedBusinessRule]`, `[determinedOrganizationUnit]`, `[plannedMission]`, `[implementingTactic]`, `[operativeVision]`, `[componentStrategy]`, `[amplifyingGoal]`, `[deliveringMission]`, `[quantifyingObjective]`, `[amplifiedVision]`, `[quantifiedGoal]`, `[categorizingInfluencerCategory]`, `[sourceInfluencingOrganization]`, `[recognizingOrganizationUnit]`, `[categorizedInfluencer]`, `[categorizingOrganizationCategory]`, `[influencingOrganizationUnit]`, `[providedInfluencer]`, `[categorizedInfluencingOrganization]`, `[identifyingAssessment]`³, `[regulatingDirective]`, `[categorizedAssessment]`.

Syntax

```
Assessment ::=  
    [motivatedDirective]::Directive+ [affectedEnd]::End+ [judgedInfluencer]::Influencer+ [assessingOrganizationUnit]::OrganizationUnit
```

```
MotivationElement ::=  
    Assessment
```

```
MotivationElement ::=  
    Means
```

```
MotivationElement ::=
```

extracted

Browsable Ada 83 Grammar



Grammar extracted by [Vadim Zaytsev](#), see the [Grammar Zoo](#) entry for details: [ada/ada83/ichbiah/extracted](#)
Source used for this grammar: Jean D. Ichbiah, *Preliminary Ada reference manual; Syntax Summary*, ACM SIGPLAN Notices, Volume 14 Issue 6a, June 1979, pages E-1 to E-5 (142-146) [[DOI](#)]

Summary

- Total **134** production rules with **245** top alternatives and **1321** symbols.
- Vocabulary: **244** = **152** nonterminals + **92** terminals + **0** labels + **0** markers.
- Total **152** nonterminal symbols: **134** defined (see below), **0** root (—), **4** top ([pragma](#), [logical_operator](#), [exponentiating_operator](#), [compilation](#)), **18** bottom ([task_name](#), [static_expression](#)⁴, [character](#), [digit](#)⁴, [subtype_name](#), [entry_name](#), [constant_name](#), [unit_name](#)², [type_name](#)⁴, [exception_name](#)², [module_name](#)², [component_name](#)², [QUOTE](#)², [character_literal](#), [subprogram_name](#), [underscore](#)³, [lower_case_letter](#), [upper_case_letter](#)).
- Total **92** terminal symbols: **63** keywords (["pragma"](#), ["constant"](#), ["type"](#)³, ["is"](#)¹³, ["subtype"](#), ["new"](#)³, ["range"](#)³, ["digits"](#), ["delta"](#), ["array"](#), ["of"](#)³, ["choice"](#), ["others"](#)², ["record"](#)⁴, ["end"](#)¹², ["null"](#)³, ["case"](#)⁴, ["when"](#)⁶, ["access"](#), ["all"](#), ["and"](#)³, ["or"](#)⁴, ["xor"](#)², ["not"](#)³, ["in"](#)⁵, ["mod"](#)², ["return"](#)², ["if"](#)², ["then"](#)³, ["elseif"](#), ["else"](#)³, ["loop"](#)², ["for"](#)⁶, ["reverse"](#), ["while"](#), ["exit"](#), ["goto"](#), ["assert"](#), ["function"](#), ["procedure"](#), ["out"](#)², ["begin"](#)³, ["exception"](#)⁵, ["declare"](#), ["private"](#)², ["package"](#), ["task"](#), ["body"](#)², ["restricted"](#)³, ["use"](#)⁶, ["renames"](#)⁴, ["initiate"](#), ["entry"](#), ["accept"](#), ["do"](#), ["delay"](#), ["select"](#)², ["abort"](#), ["separate"](#)³, ["raise"](#), ["generic"](#), ["packing"](#), ["at"](#)³), **0** numerics (—), **28** signs (["#"](#), ["."](#)⁶, ["+"](#)³, ["-"](#)³, ["\("](#)¹⁸, ["\)](#)¹³, ["\)](#)¹⁸, [";](#)⁴⁷, [":"](#)⁵, [":="](#)⁴, [".."](#)², ["|"](#)⁴, ["=>"](#)⁶, ["'"](#), ["##"](#)², ["="](#), ["/="](#), ["<"](#), ["<="](#), [">"](#), [">="](#), ["&"](#), ["@"](#), ["/"](#), ["<<"](#), [">>"](#), ["::"](#), [":::"](#)).

Syntax

```
identifier ::=  
    letter (underscore? letter_or_digit)*
```

```
letter_or_digit ::=  
    letter  
    digit
```

extracted

Browsable Ada 83 Grammar



Grammar connected by [Vadim Zaytsev](#), see the [Grammar Zoo](#) entry for details: [ada/ada83/ichbiah/connected](#)
Source used for this grammar: Jean D. Ichbiah, *Preliminary Ada reference manual; Syntax Summary*, ACM SIGPLAN Notices, Volume 14 Issue 6a, June 1979, pages E-1 to E-5 (142-146) [[DOI](#)]

Summary

- Total **134** production rules with **243** top alternatives and **1335** symbols.
- Vocabulary: **243** = **141** nonterminals + **92** terminals + **0** labels + **10** markers.
- Total **141** nonterminal symbols: **134** defined (see below), **1** root ([compilation](#)), **1** top ([pragma](#)), **7** bottom ([character](#), [digit](#)⁴, [DQUOTE](#)², [character_literal](#), [underscore](#)³, [lower_case_letter](#), [upper_case_letter](#)).
- Total **92** terminal symbols: **63** keywords (["pragma"](#), ["constant"](#), ["type"](#)³, ["is"](#)¹³, ["subtype"](#), ["new"](#)³, ["range"](#)³, ["digits"](#), ["delta"](#), ["array"](#), ["of"](#)³, ["choice"](#), ["others"](#)², ["record"](#)⁴, ["end"](#)¹², ["null"](#)³, ["case"](#)⁴, ["when"](#)⁶, ["access"](#), ["all"](#), ["and"](#)², ["or"](#)³, ["xor"](#), ["not"](#)³, ["in"](#)⁵, ["mod"](#)², ["return"](#)², ["if"](#)², ["then"](#)³, ["elseif"](#), ["else"](#)³, ["loop"](#)², ["for"](#)⁶, ["reverse"](#), ["while"](#), ["exit"](#), ["goto"](#), ["assert"](#), ["function"](#), ["procedure"](#), ["out"](#)², ["begin"](#)³, ["exception"](#)⁵, ["declare"](#), ["private"](#)², ["package"](#), ["task"](#), ["body"](#)², ["restricted"](#)³, ["use"](#)⁶, ["renames"](#)⁴, ["initiate"](#), ["entry"](#), ["accept"](#), ["do"](#), ["delay"](#), ["select"](#)², ["abort"](#), ["separate"](#)³, ["rause"](#), ["generic"](#), ["packing"](#), ["at"](#)³), **0** numerics (—), **28** signs (["#"](#), ["."](#)⁶, ["+"](#)³, ["-"](#)³, ["\("](#)¹⁸, [" "](#)¹³, ["\)"](#)¹⁸, [";"](#)⁴⁷, ["."](#)⁵, ["="](#)⁴, [".."](#)², ["|"](#)⁴, ["=>"](#)⁶, ["'"](#), ["="](#), ["/="](#), ["<"](#), ["<="](#), [">"](#), [">="](#), ["&"](#), ["*"](#), ["/"](#), ["<<"](#), [">>"](#), ["="](#), ["::="](#), ["***"](#)).
- Total **10** markers: [<type_name>](#)⁴, [<subtype_name>](#), [<constant_name>](#), [<component_name>](#)², [<subprogram_name>](#), [<unit_name>](#)², [<module_name>](#)², [<task_name>](#), [<entry_name>](#), [<exception_name>](#)².

Syntax

```
identifier ::=  
    letter (underscore? letter_or_digit)*
```

```
letter_or_digit ::=  
    letter  
    digit
```

connected


```
export BGF |home:///projects/webzoo-prep/zoo/ada/ada83/ichbiah/extracted/grammar.bgf|.
reroot to compilation.
fold exponentiating_operator everywhere.
eliminate logical_operator.
// As described in §4.8, a static_expression can be detected at compile time and
must only contain
// (a) literals
// (b) aggregates whose components are static expressions
// (c) constants initialised by static expressions
// (d) predefined operators, functions and attributes
// (e) qualified static expressions
// (f) indexed and selected components of constants
// There are two ways to solve this: either duplicate a grammar fragment and make
these changes or use the easy way:
define static_expression ::= expression ; .
// This is a convention they used (italics)_(name) to mean (name) with some
annotation
replace task_name with <task_name>:name everywhere.
replace subtype_name with <subtype_name>:name everywhere.
replace entry_name with <entry_name>:name everywhere.
replace constant_name with <constant_name>:name everywhere.
replace unit_name with <unit_name>:name everywhere.
replace type_name with <type_name>:name everywhere.
replace exception_name with <exception_name>:name everywhere.
replace module_name with <module_name>:name everywhere.
replace component_name with <component_name>:name everywhere.
replace subprogram_name with <subprogram_name>:name everywhere.
export BGF |home:///projects/webzoo-prep/zoo/ada/ada83/ichbiah/connected/grammar.bgf|.
```

Maturity Model



- fetched
- extracted
- connected
- adapted
- exported



?

fetches

extracts

connects

adapts

exports

?

copy

download

git

scan

type over

fetches

recovery

automatic

retype

extracts

recovery

semi-automated

connects

semi-automated

mutation

adapts

putback

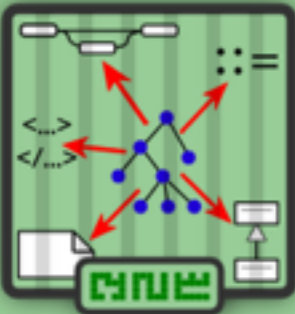
normalise

pretty-print

exports


Grammar Zoo

file:///Users/zaytsev/projects/zoo/web/index.html



GrammarZoo

The objective of the Grammar Zoo is to accumulate grammars of various software languages, extracted and recovered from language documentation, parser specifications and other artefacts and make them available in a range of formats.



483 entries and counting

772 grammars: 483 fetched + 260 extracted + 20 connected + 7 corrected + 1 recovered + 1 imported

Bulk download of the whole corpus:

[Ada](#) — [API](#) — [Assembly](#) — [ATL](#) — [AWK](#) — [Basic](#) — [C](#) — [C++](#) — [C#](#) — [Dart](#) — [DSL](#) — [Eiffel](#) — [Fortran](#) — [HTML](#) — [Java](#) — [JavaScript](#) — [Logo](#) — [Markup](#) — [Meta](#) — [ML](#) — [Modula](#) — [Occam](#) — [Ontology](#) — [Pascal](#) — [Python](#) — [UML](#) — [XMLware](#)

(●●●) [Ada](#) (↑)

[Ada 83](#) — [Ada 95](#) — [Ada 2005](#)

(●●●) [Ada 83](#) [git]

(●●●) [Ichbiah](#) [git] [ReadMe]

- Source: Jean D. Ichbiah, *Preliminary Ada reference manual; Syntax Summary*, ACM SIGPLAN Notices, Volume 14 Issue 6a, June 1979, pages E-1 to E-5 (142-146) [DOI]
- (→) The **fetched** grammar is [src.syntax.summary.txt]
- `dir: fetched` `level: 0` `method: copy`
- (→) The **extracted** grammar is [grammar.bgf] [Browse now!]
- Files used: [connect.glue]
- `of: Ada 83` `dir: extracted` `level: 1` `method: typeover` `fileused: connect.glue` `toolused: grammarlab`
- (→) The **connected** grammar is [grammar.bgf] [Browse now!]
- Files used: [connect.glue]
- `of: Ada 83` `dir: connected` `level: 2` `method: semi-automated` `fileused: connect.glue` `toolused: grammarlab`

(●) [LNCS 0106](#) [git] [ReadMe]

- Source: Proposed Standard Document United States Department of Defense, *The Programming Language Ada Reference Manual*, 1981, Appendix E: Syntax Summary, pages 221–225 [DOI]
- (→) The **fetched** grammar is [src.syntax.summary.txt]
- `dir: fetched` `level: 0` `method: copy`

(●) [LNCS 0155](#) [git] [ReadMe]

- Source: ANSI/MIL-STD-1815A-1983, *The Programming Language Ada Reference Manual*, 1990, Appendix E: Syntax Summary, pages E-1–E-6 (277–282) [DOI]
- (→) The **fetched** grammar is [src.syntax.summary.txt]

Conclusion

- More details?
 - read the paper (ME'14, pages 42–51)
 - read the other paper (“Grammar Zoo”, SCP, 2014)
- Grammar Zoo
 - new version SOON
- Sources
 - All photos taken from PEXELS, CCO Universal license.
- Questions?