Praśnaḥ

Aṣṭādhyāyī 2.0 Query Tool

USER MANUAL

Preface

Aṣṭādhyāyī 2.0 is a digital, interactive and searchable edition of Aṣṭādhyāyī, the famous treatise about Sanskrit morphology, syntax and semantics, written in the 4th century BCE by the ancient Indian grammarian Pāṇini. The search feature of Aṣṭādhyāyī 2.0 is provided by Praśnaḥ (English: question or inquiry), a dedicated web application, that is also referred to as "query tool" in this document. Praśnaḥ provides a lightweight and powerful query language to search for sūtras and its translations as well as grammatical attributes such as case, number and gender of their textual components. In addition, the query tool can be used to search for linguistic annotations such as word-semantic meanings, topics, lexemes and, moreover, to ascertain whether components are declinable. Queries on textual components can also be expanded to include anuvṛtti. The query language used by the Aṣṭādhyāyī 2.0 Query Tool allows to perform queries by the use of predefined query terms, logical operators, wildcards, parentheses, transliterations as well as special operators to specify component or non-component based searches. The latter two can also be linked by the use of logical operators and can be weighted by means of bracketing.

This manual highlights all features of the Aṣṭādhyāyī 2.0 Query Tool and helps to understand the key principles of the query language. In addition, sample searches and hints are given throughout for a successful learning experience. Be advised that this manual does not include the complete language reference. Hence you are encouraged to refer to the quick reference provided on the Aṣṭādhyāyī 2.0 website.

Prerequisites

The Aṣṭādhyāyī 2.0 Query Tool uses HTML5 and CSS3 technology and should therefore support a broad range of commonly used web browsers. For best user experience we recommend using a web browser which has been tested for compatibility and performance with our web application such as the latest versions of Firefox, Google Chrome and Safari. Also note that the query tool requires a display resolution of at least 1280 by 1024 to be fully functional. However for a better perception of the query tool, it is strongly recommended to have a higher resolution such as 1400 by 1050. Be advised that in order to use the

Aṣṭādhyāyī 2.0 Query Tool, it is absolutely necessary that your browser has JavaScript and Cookies enabled. In case these are disabled due to privacy concerns, the upshot will be that the web application refuses to work and will output an error message (Fig.1) (Fig. 2).

NOTE: JavaScript and Cookies are solely used to enhance usability and to provide extended features without sacrificing privacy at any time.

USER INTERFACE

The user interface of the Aṣṭādhyāyī 2.0 Query Tool follows a minimalistic design approach to emphasize on usability. Hence the interface has been divided into three sections which are explained below.

Navigation Bar

The horizontal navigation bar at the top of the screen allows quickly switching to different parts of the Aṣṭādhyāyī 2.0 website (Fig. 3).

Side Panel

The side panel on the left of the horizontal bar shows the current version of the Aṣṭādhyāyī 2.0 Query Tool and indicates the user's unique identification code[1]. It furthermore provides the latest release notes accommodating new features as well as contemporary issues (Fig. 4).

Additionally, the side panel has a toggle to cycle through a set of predefined color schemes (Fig. 5), allowing the user to set a color which achieves the best possible screen reproduction.

Query Panel

The query panel on the right of the horizontal bar is the actual search feature of Aṣṭādhyāyī 2.0. The panel consists of three boxes, whereas the sole purpose of the top or about box is to provide a brief description of the Aṣṭādhyāyī 2.0 Query Tool (Fig. 5).

Right below the about box is the query box which allows users to type in or rephrase their queries. The query box contains a toggle labeled “Example” which can be left clicked to cycle through a list of predefined sample searches (Fig.6).

Beneath the query box are four buttons labeled “submit query”, “clear query”, “anuvrtti” and “show help”(from left to right) (Fig. 7), which are described later. The information box (Fig. 8) at the very bottom notifies the user in case of an error or malformed query with a corresponding message.

CONDUCTING A SEARCH

In order to conduct a search the user has to navigate to the Aṣṭādhyāyī 2.0 Query Tool by left-clicking the “search” button on the navigation bar at the top of the screen. Once the Aṣṭādhyāyī 2.0 Query Tool is loaded the caret can be revealed and placed into its position by clicking anywhere inside the query box. Assuming that the user has not previously used the Aṣṭādhyāyī 2.0 Query Tool the query box will be empty.

[1] The purpose of the unique identification code is explained in detail on the next page.

HINT: Clicking the “Example” toggle will cycle through a list of predefined queries which can be used as a template to conduct a new query expression.

Otherwise the box will reveal the user’s last query which can then be rephrased. In order to start a new search from scratch, it is recommended to clear the query box by simply clicking on the “clear query” button. Once the query has been conducted or rephrased, the search can be performed by clicking the “submit query” button. To search for anuvṛtti you need to make sure that anuvṛtti has been enabled by clicking on the anuvṛtti button.

HINT: Instead of clicking the “submit query” button to run a search, the user can also press the CTRL and ENTER key simultaneously. This might be a great time saver when running multiple searches or rephrasing specific query expressions.

Once a search has been submitted the user will be redirected to the Aṣṭādhyāyī 2.0 Browser which will display the results and number of hits. In order to rephrase the previous query the user may click on the “search” button on the navigation bar which will redirect him to the Aṣṭādhyāyī 2.0 Query Tool. This procedure can be repeated until the user obtains the results he expects.

HINT: After submitting a search it’s very easy to switch back and forth between the search results and the Aṣṭādhyāyī 2.0 Query Tool by pressing the backspace key. This keyboard shortcut is supposed to work in all major web browsers.

In order to improve the readability of query expressions it’s possible to add line breaks by pressing the ENTER or RETURN key. Note that the query box uses line wraps, which will automatically places statements on a new line before they reach the end of the query box. A line wrap however must not be confused with intentional line breaks.

HINT: Comments may be used to increase the readability or to add notes to query expressions by using the pound key. Everything from the pound to the end of the line is regarded as a comment and thus being ignored by the query tool’s interpreter.

For the sake of functionality, the user’s previous query is stored in our database. In order to link a query to a specific user, the Aṣṭādhyāyī 2.0 Query Tool makes use of a unique identification string which is stored on the user’s computer as well as in our database. Note that once the user clears the browser cache the Aṣṭādhyāyī 2.0 Query Tool can no more display the user’s last query expression. Additionally it may also be worth knowing that the user’s last query has a validity of 24 hours after which it will be automatically deleted.

In order to use the Aṣṭādhyāyī 2.0 Query Tool to its full potential you should continue reading on the following section, which will explain everything you need to know about the query language.

The Query Language

The Aṣṭādhyāyī 2.0 Query Tool provides a lightweight and powerful query language to conduct searches. In order to successfully formulate queries, it is essential to have a good understanding of the query language itself. The query language is subdivided into several language elements or topics for that matter, including query terms, logical operators, order of operations, pattern matching, transliterations, non-component searches and search restrictions which are described in detail in the following paragraphs.

However, before starting out it is imperative to emphasize the meaning of sūtras, components as well as linguistically annotations in the context of to the Aṣṭādhyāyī 2.0 Query Tool. A sūtra denotes a distinct type of literary composition written in one single line, which is made up of words to which we will hereinafter refer to as components. These components have grammatical attributes (i.e. case, number, gender) which can be searched for in a regular or non-component based search, respectively. The results will be one or more sūtras each containing components whose attributes matched the query expression.

Query Terms

The Aṣṭādhyāyī 2.0 Query Tool uses predefined query terms for searching Aṣṭādhyāyī 2.0. Each of these query terms denotes a specific field of search and can be used repeatedly in a query expression. Even though query terms and logical operators are case-insensitive, it is generally advised to use capital letters for both to emphasize readability of complex query expressions. Also note that all query terms are meant to search for component's attributes except for the terms "SUTRA" and "TRANS". The latter two are solely used for searching sūtras and its corresponding translations.

A query term has consistently to be followed by its value which must be surrounded by parentheses. For better readability though, one may use whitespaces or even line breaks in-between query terms and values. Also note that when using the query terms "CAS", "GEN", "GRAM" or "NUM" you must choose one of their predefined values and write them in lowercase letters. Furthermore, it is worth knowing that certain query terms and values offer alternative spellings.

A common query might be the search for sūtras to which one or more components have a certain grammatical declension. For example searching for sūtras for which there are components having either feminine, singular or nominative form, you would use the expression: "GEN (fem) AND NUM (sg) AND CAS (nom)".

Here the query term "GEN" defines the grammatical gender to be feminine whereas the query term "NUM" defines the grammatical number to be singular and "CAS" finally defines the component to be written in nominative case. Note that all three terms are linked with the logical "AND" operator to force the search for sūtras containing components where each of them has to meet at least one of those three grammatical attributes mentioned before. Another useful query might be the search for sūtras containing one or more components which are compositions or indeclinable, by using the expression: "GRAM (comp) OR GRAM (indecl)".

Searching for the semantically meaning and lexeme in components can be achieved by using the query terms "MEAN" and "LEX", whereas the search for sūtras and its translations itself is being accomplished by using the terms "SUTRA" and "TRANS", respectively. It is important to understand that the form always remains the same, irrespective of the query terms actually being used. For a complete list of query terms and predefined values you're encourage to have a look at the quick reference.

Logical Operators

In all cases where query expressions have been composed with more than one term or compound expression, it is necessary to link those using logical operators. For that matter the Aṣṭādhyāyī 2.0 Query

Tool supports the three commonly used operators “AND”, “OR” and “NOT” to indicate conjunctions, disjunctions or negations, respectively. All of them can be combined in one single or compound expression repeatedly.

NOTE: Logical operators are also case-insensitive but it is advised to use capital letters to improve readability of your query expressions.

A logical conjunction is denoted by the “AND” operator and may only return a search result if and only if either query term(s) or compound expression(s) are true. For instance the expression “CAS (nom) AND GEN (f)” will only return sūtras containing one or more components of which at least one has the grammatical case in nominative form and one (eventually the same) the grammatical gender in feminine form.

A logical disjunction however, which is denoted by the “OR” operator returns a search result only if either or both query term(s) or compound expression(s) are true. This means that the expression “CAS (num) OR GEN (f)” for instance, will return sūtras containing one or more components of which at least one has the grammatical case in nominative form or sūtras containing one or more components having a the grammatical gender in feminine form.

NOTE: Whenever you use both logical operators “AND” and “OR” you need to specify the precedence of operation (see next paragraph).

In order to negate values you have to use the “NOT” operator which has to precede any query term and value that should be excluded from search.

NOTE: The “NOT” operator may not precede compound expressions but instead has to precede any query term whose value(s) you may like to negate. Also be advised that query expressions may not begin with a logical operator after all.

Order of Operations

Brackets are used to specify the order of operations of query terms and are a decisive factor when conducting queries. The Aṣṭādhyāyī 2.0 Query Tool uses two different kinds of parentheses. At the first level square brackets “[]“ are used to enclose two or more statements, as with the expression “CAS (num) OR [GEN (f) AND NUM (sg)]”. Since square brackets may not be nested the user has to use curly braces “{}” at the second level of bracketing such as “CAS (acc) OR { CAS (num) OR [GEN (f) AND NUM (sg)] }”. Be advised that bracketing of nested or compound expressions is not allowed. Also note that component based searches may use the same bracketing and are not regarded as nested expressions.

NOTE: When using the “AND” and “OR” operators in your expression you must always specify the order of operation.

Pattern Matching

The Aṣṭādhyāyī 2.0 Query Tool allows the use of the wildcard characters “*” and “?” to pattern matching values. Both wildcards can be placed at any point within a query term’s value. Note that the asterisks “*”matches any character one or more times whereas the question mark “?” matches a single character only once.

NOTE: The query terms „SUTRA“ and „TRANS“ both use the asterisk pattern character at the beginning and end of your search strings, to match any sūtra or translation in which the given search term appears.

Transliterations

Since sūtras in Sanskrit are written in the Devanagari script, the Aṣṭādhyāyī 2.0 Query Tool allows the use of transliteration, which in regard to Aṣṭādhyāyī 2.0, is the conversion of Roman script to Devanagari. The transliteration can be achieved by the use of Velthuis and may only be used in conjunction with the query terms "SUTRA", "MEAN", "TRANS" and "LEX". Also note that transliterations must be all written in lowercase and be surrounded by forward slashes "/ /". A short example to illustrate the use of transliteration would be the expression "SUTRA (*/aa/*)", which searches for sūtras containing the short vowel "अ" in Devanagari script. However for a complete list of transliterations you are strongly encouraged to read the quick reference.

Component Searches

When using the Aṣṭādhyāyī 2.0 Query Tool, searches are usually based on linguistic attributes of spanned components. However when searching for sūtras and translations you may conduct a strict component based search by surrounding the expression with right and left angle brackets "< >", as in "<LEX (kart/.r/) AND CAS (loc) >". The query shows all sūtras in which the Sanskrit term for agent "(kart/.r/)" is mentioned in locative case. Whereas in a regular or spanned component search one would conduct a query like "LEX (kart/.r/) AND CAS (loc)", which would search for all sūtras in which the Sanskrit term for agent "(kart/.r/)" is mentioned and where a component (possibly different from "(kart/.r/)") is in locative case.

NOTE: Component and non-component based searches can be linked using logical operators. Additionally a non-component based search may use all kinds of bracketing irrespective of bracketing used by the outer expression.

Search Restrictions

The Aṣṭādhyāyī 2.0 Query Tool allows you to narrow down your search on sūtras or components from a certain book and chapter of Aṣṭādhyāyī. This means you can specify a range of sūtras in which to search in and might come in handy in case that you already know in what sūtras or components you're actually interested in. The special operator needed is "SNR" and can denote up to five ranges which have to be separated by a semi-colon.

NOTE: The "SNR" search restriction may only be used once per query and effects the entire expression.

For instance when searching for sūtras for which there are components having either feminine, singular or nominative form only in the first and second book, you would use the expression: "[GEN (fem) AND NUM (sg) AND CAS (nom)] and SNR (1-2)". This query terms can be negated with the "NOT" operator one could also conduct the same query using the expression: "[GEN (fem) AND NUM (sg) AND CAS (nom)] and NOT SNR (3-8)". Be advised that when searching within one specific book the user still has denote it as a range (i.e. SNR (1-1)). Searching in two chapters from the same book can be achieved by specifying ranges, as in "SNR (1.2-1.2;1.4-1.4)". Here the search would be conducted only on Chapter two and four from the first book of Aṣṭādhyāyī.

Appendix: Quick Reference

This quick reference provides a summary of query terms, search operators and transliterations used by the Aṣṭādhyāyī 2.0 Query Tool.

Query Terms

Query Term	Alt. Spelling	Search Type	Short Description	Value(s) / Alt. Spell.	Example(s)
cas	casus case	component	grammatical case	nom; nominative gen; genitive loc; locative abl; ablative acc; accusative dat; dative voc; vocative	cas(nom)
gen	genus gender	component	grammatical gender	m; masculine f; feminine n; neuter	gen(f)
gram	grammar	component	grammatical type	comp; composition decl; declined indecl; undeclined	gram(decl)
lex	lexeme	component	lexeme	*[2]	lex(nose)
mean	meaning sem semantic	component	semantic	*	mean(nose)
num	numerus number	component	grammatical number	sg; singular du; dualis pl; plural	num(sg)
sutra[3]	-	sutra	sutra	*	sutra(ion)
topic	-	component	topic	*	
trans	translation	sutra	Translation	*	trans(nose)

Search Operators

Operator	Short Description	Value(s)	Example(s)
and	logical conjunction resulting in true if both of its operands are true, otherwise the value is false	-	cas(nom) and gen(f)
or	logical disjunction resulting in true if one or both operands are true, otherwise the value is false	-	num(sg) or trans (mean)
Operator	Short Description	Value(s)	Examples(s)
not	negates the result of the following query term	-	not gram(decl)

snr	specifies the range of books, chapters and sutras to search in	x-x
		x.x-x.x
			snr(1-4)
			snr(1.3.3-2)

[2] Values marked with an asterisk may be arbitrary.

[3] The query terms „sutra” and „trans“ both use the asterisk character by default to match any sutra or translation in which the given search term appears.

		x.x.x-x.x.x	snr(1.2-1.4;4-5)
*	matches any character one or more times	-	mean(nos*)
?	matches a single character one time	-	mean(n?se)
<>	indicates a component based search	-	< mean (nose) and gen (n) >
()	parentheses enclose values	-	sutra(ion)
[]	square brackets enclose multiple expressions	-	cas(nom) and [gen(f) or gen(n)]
{}	curly brackets enclose expressions in brackets	-	{ cas(nom) and [gen(f) or gen(n)] } or num(du)

Transliterations

Operator	Description	Velthuis	IAST	Devanagari	Example(s)

//	forward slashes enclose transliterations	a	a	अ	sutra(/.rr/)
					sutra(e/.d/ion)
					sutra(i/.l/io/~n/)

		aa	ā	आ	
		i	i	इ	
		ii	ī	ई	
		u	u	उ	
		uu	ū	ऊ	
		.r	ṛ	ऋ	
		.rr	ṝ	ॠ	
		.l	ḷ	ऌ ḷ	
		.ll	ḹ	ॡ ḹ	
		e	e	ए	
		ai	ai	ऐ	
		o	o	ओ	
		au	au	औ	
		.m	ṃ	अं	
		.h	ḥ	अः	
		~	‘	ऽ	
		ka	ka	क	
		kha	kha	ख	
		ga	ga	ग	
		gha	gha	घ	
		“na	ṅa	ङ	
		ca	ca	च	
		cha	cha	छ	
		ja	ja	ज	

		jha	jha	झ	
		~na	ña	ञ	
		.ta	ṭa	ट	
		.tha	ṭha	ठ	
		.da	ḍa	ड	
		.dha	ḍha	ढ	
		.na	ṇa	ण	
		ta	ta	त	
		tha	tha	थ	
		da	da	द	
		dha	dha	ध	
		na	na	न	
		pa	pa	प	
		pha	pha	फ	
		ba	ba	ब	
		bha	bha	भ	
		ma	ma	म	
		ya	ya	य	
		ra	ra	र	
		la	la	ल	
		va	va	व	
		“sa	śa	श	
		.sa	ṣa	ष	
		sa	sa	स	
		ha	ha	ह	