

**Version:** 1.0  
**Date:** March 12th 2003

# **The Monet Mathematical Query Ontology**

Stephen Buswell, Phillip Lord

**Deliverable D13 (Public)**

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Anatomy of a Query</b>	<b>2</b>
2.1	The Mathematical Part . . . . .	2
2.2	The Logistical Part . . . . .	2
<b>3</b>	<b>Anatomy of a Response</b>	<b>3</b>
<b>4</b>	<b>General Comments</b>	<b>3</b>

## 1 Introduction

The mathematical query ontology describes a general query language in which mathematical queries will be submitted to a MONET broker. In addition to the mathematical problem itself (or a description of it), the query can contain additional information, either about the client, or extra constraints on the solution, or constraints about the selection of the services which might provide the solution.

The query is sent to the broker which will return a (possibly ordered) list of services and/or execution plans which can answer the query.

## 2 Anatomy of a Query

A query consists of two distinct parts:-

**A mathematical part** A description of the mathematical problem expressed in MSDL

**A logistical part** Additional constraints on the query. This is to support arbitrary representations of the constraints, which are not defined here, but are potentially interpretable by the broker. For example, constraints may be expressed in an RDF vocabulary.

The main purpose of the query ontology is to provide a simple wrapper for a number of different constraints which together comprise a query, and for the responses to this query. In one sense the mathematical part of the query could be considered to be just another constraint. However, it is identified specifically here, both to make explicit the representation of this part of the query, and also because, unlike the logistical part of the query, it is mandatory in any query.

### 2.1 The Mathematical Part

The mathematical part of a query is expressed in terms of the “classifications” section of the MSDL schema. This may include

- A reference to a problem description library. This may be a generic problem description defined using MPDL, or a specific instance of a generic problem, supplying values for the input parameters.
- References to problem taxonomies
- A reference to the required semantics of the service requested, for example by reference to OpenMath CDs or MathML semantics
- A reference to a semantic description in an appropriate framework such as RDF
- A list of supported directives (find, prove, etc.)

### 2.2 The Logistical Part

The Logistical part can include

- Arbitrary constraints expressed as metadata in RDF(S). This may match RDF metadata in the service description or third party information (from a UDDI registry for example).
- Preferences, for example geographical/logical locations of service instances, blacklisted services.
- Quality of Service (QOS) requirements. These may possibly be expressed in some external formalism such as in GRID GESA.
- Client-side information for use in query resolution (e.g. PGP key).

Each constraint carries an ‘encoding’ attribute indicating the encoding formalism to the broker. This would typically be a URI identifying a namespace. It is not expected that every broker understands every formalism, however some, for example an RDF encoding of Dublin Core data, may be widely understood. Furthermore, each constraint carries a ‘weight’ representing the importance of the constraint from the client viewpoint. Typical values include: “required”; “excluded”; “preferred”; “optional”; ... Note that there is no general solution to the ordering problem which would allow a broker to express a ranking in the presence of multiple independent constraints with values other than “required” or “excluded”.

### 3 Anatomy of a Response

A response consists of two distinct parts:-

**A general part** General information indicating the success of the query, the number of solutions proposed and the understandability of any constraints from the viewpoint of the broker.

**A solutions part** For each solution, the solution type (service or execution plan) and the bindings allowing access to the solution are given. The applicability of each constraint to this solution is stated. Metadata can be included to clarify the constraint applicability. If the solution solves a related but different problem (for example a subset of the problem as stated), then the MSDL for the problem actually solved is given. If absent, the service offers a complete solution to the service as requested.

### 4 General Comments

Note that this request/response mechanism does not cater for the execution of the service by the broker on behalf of the client, although it could be extended to handle this.