```
================================================================================
biocompute object example for "antiviral_resistance_detection[ICD:B17]"
================================================================================
{
    "id": "obj.1001",
    "type":"antiviral_resistance_detection[ICD:B17]",
    "name": "HCV1a[taxID:31646] ledipasvir[PubChem:67505836] resistance SNP [SO:
        0000694]",
    "version": "1.2",
    "verification": "manual",
    "creation_date": "Mon Jun 6 13:25:21 EDT 2016",
    "modification_date": "Thu Jun 9 08:12:14 EDT 2016",
    "publication_status": "draft",
    "keywords": [
        "antiviral resistance",
        "SNP"
    ],
    "xref": [
        "DO:1883",
        "ICD:B17",
        "taxID:9606"
        ],
    "usability_domain": [
        "Identify baseline single nucleotide polymorphisms (SNPs[SO:0000694]),
            insertions[SO:0000667], and deletions[SO:0000045] that correlate with
            reduced ledipasvir[PubChem:67505836] antiviral drug efficacy in
            Hepatitis C virus subtype 1",
        "Identify treatment emergent amino acid substitutions[SO:0000048] that
            correlate with antiviral drug treatment failure",
        "Determine whether the treatment emergent amino acid substitutions[SO:
            0000048] identified correlate with treatment failure involving other
            drugs against the same virus"
    ],
    "authors": [
        {
                "first_name": "Charles",
                "last_name": "Darwin",
                "affiliation": "HMS Beagle"
        },
        {
                "first_name": "Rosalind",
                "last_name": "Franklin",
                "affiliation": "King's College, London"
        }
    ],
    "pipeline": "obj.10001",
    "alignment" : {
        "seed": 14,
        "minimum_match_len": 50,
        "divergence_threshold_percent": 10
        },
    "variant_calling" : {
```

```
            "minimum_coverage": 20,
            "frequency_cutoff": 0.20
            },
        "verification_kit":{
            "reference_sequence":[
                {
                    "source": "NCBI",
                    "db": "genbank",
                    "id": ["JQ434001.1"]
                },
                {
                    "source": "HIVE",
                    "db": "Honeycomb",
                    "id": "obj.00123"
                }
            ],
            "sequence_reads": [
                {
                    "source": "HIVE",
                    "db" : "Honeycomb",
                    "id": "obj.00013"
                },
                {
                    "source": "HIVE",
                    "db": "Honeycomb",
                    "id": "obj.10065"
                }
            ],
            "snv_profile_file": {
                "format": [".csv",".vcf"],
                "filename": "SNPProfile"
            },
            "alignment_file": {
                "format": [".sam"],
                "filename": "SNPProfile.sam"
            },
            "aminoacid_calls_file": {
                "format": [".csv",".vcf"],
                "filename": "AAcalls.csv"
            }
        }
}


================================================================================
pipeline definition example for "HIVE-Viral Mutation Detection Pipeline"
================================================================================
{
    "_id": "obj.10001",
    "_type": "pipeline_definition",
    "name": "HIVE-Viral Mutation Detection Pipeline",
```

```
    "description": "HIVE-Viral Mutation Detection uses two HIVE tools: Hexagon
            for alignment and Heptagon for variant calling. Only the options that
            need  to be selected or changed are listed.",
    "workflow": {
        "launcher": "system",
        "script": "obj.123123"
    },
    "tool_used": [
        {
            "name": "HIVE-hexagon",
            "version": "1.3",
            "arguments": [
                {
                    "parameter": "seed",
                    "value": 16
                },
                {
                    "parameter": "divergence_threshold_percent",
                    "value":  15
                },
                {
                    "parameter": "minimum_match_len",
                    "value":  100
                }
            ]
        },
        {
            "name": "HIVE-heptagon",
            "version": "1.1",
            "arguments": [
                {
                    "parameter": "min_coverage",
                    "value": 10
                },
                {
                    "parameter": "freq_cutoff",
                    "value": 0.10
                }
            ]
        }
    ]
}


================================================================================
typedef for "antiviral_resistance_detection"
================================================================================
{
    "_type": "type",
    "name": "antiviral_resistance_detection",
    "description": "This pipeline uses an aligner and then a profiler to detect
            possible mutations involved in drug resistance.",
    "_fields": {
```

```
"reference_sequence": {
    "description": "reference sequences can be either FASTA from known
        sources or identifiers from HIVE database. These must be saved as
        a 'genome' type in HIVE to use in this field",
    "title": "Reference Sequences",
    "_type": "xref",
    "_role": "input",
    "_vital": "true",
    "_plural": "true"
},
"sequence_reads": {
    "description": "sequences can be either FASTA from experimental
        sources or identifiers from HIVE database. These must be saved as
        a 'nuc_read' type in HIVE to use in this field",
    "title": "Nucleotide Read",
    "_type": "xref",
    "_role": "input",
    "_vital": "true",
    "_plural": "true"
},
"alignment": {
    "description": "Arguments for the alignment, HIVE Hexagon",
    "title": "HIVE Hexagon Alignment Engine",
    "seed": {
        "_type": "integer",
        "_default": 11,
        "_limit": {
            "eval-js" : {"$seed>8 && $seed<12"}
        }
    },
    "minimum_match_len,": {
        "_type": "integer",
        "_default": 75
    },
    "divergence_threshold_percent": {
        "_type": "real",
        "_default": 15
        "_limit:" {
            "eval-js" : {"$divergence_threshold_percent>10 &&
                    $divergence_threshold_percent<40"}
        }
    }
},
"variant_calling": {
    "description": "arguments for the engine to perform coverage
        analysis, base-calling, SNP calling",
    "title": "Sequence Profiling Engine",
    "minimum_coverage": {
        "_type": "integer",
        "_default": 10,
        "_limit": {
            "eval-js" : {"$minimum_coverage>=0"}
```

```
                }
            },
            "frequency_cutoff": {
                "_type": "real",
                "_default": 0.10,
                "_limit": {
                    "eval-js" : {"$frequency_cutoff>=0 &&
                            $frequency_cutoff<=1.0"}
                }
            }
        },
        "snv_profile": {
            "description": "The output file containing the SNP Profile",
            "title": "SNPProfile Output File",
            "_role": "output",
            "_type": "outfile"
        },
        "alignment_file": {
            "description": "The resulting alignment file from the Hexagon
                computation",
            "title": "Alignment File",
            "_type": "outfile"
        },
        "aminoacid_calls": {
            "description": "the amino acid calls from Heptagon",
            "title": "Amino Acid Calls",
            "_role": "output",
            "_type": "outfile"
        }
    },
    "_inherits": [
        "base_biocompute_type"
    ]
}


================================================================================
typedef for "pipeline_definition_type"
================================================================================
{
    "_id": "obj.10001",
    "_type": "pipeline_definition",
    "name": "HIVE-Viral Mutation Detection Pipeline",
    "description": "HIVE-Viral Mutation Detection uses two HIVE tools: Hexagon
            for alignment and Heptagon for variant calling. Only the options that
            need to be selected or changed are listed.",
    "workflow": {
        "launcher": "system",
        "script": "obj.123123"
    },
    "tool_used": [
        {
            "name": "HIVE-hexagon",
```

```
                "version": "1.3",
                "arguments": [
                    {
                        "parameter": "seed",
                        "value": 16
                    },
                    {
                        "parameter": "divergence_threshold_percent",
                        "value":  15
                    },
                    {
                        "parameter": "minimum_match_len",
                        "value":  100
                    }
                ]
            },
            {
                "name": "HIVE-heptagon",
                "version": "1.1",
                "arguments": [
                    {
                        "parameter": "min_coverage",
                        "value": 10
                    },
                    {
                        "parameter": "freq_cutoff",
                        "value": 0.10
                    }
                ]
            }
        ]
    }
}


================================================================================
typedef for "base_biocompute_type"
================================================================================
{
    "_type": "type",
    "name": "base_biocompute_type",
    "description": "Provides the base fields to populate in a biocompute object",
    "_fields": {
        "validation": {
            "description": "Describes the position of an object in the validation
                process. The unreviewed flag indicates that the object has been
                submitted, but no further curation has occurred. The in_progress
                flag indicates that validation is underway. manual indicates that
                the object has been manually validated. The suspended flag
                indicates an object that was once valid is no longer considered
                valid. The error flag indicates that an error was detected with
                the biocompute object.",
            "_type": "string",
            "_limit": {
```

```
                    "choice": [
                        "unreviewed",
                        "in_progress",
                        "manual",
                        "suspended",
                        "error"
                    ]
                }
            },
            "usability_domain": {
                "description": "Provides a space for the author to define the
                    usability
                    domane of an object.",
                "_type": "string",
                "_vital": "true",
                "_plural": "true"
            },
            "pipeline" : {
                "description": "defines the fields that have to be populated for a
                    tool description in the pipeline of a biocompute object",
                "_type": "pipeline_definition"
            }
        },
        "_inherits": ["base_metadata_type"]
}


================================================================================
typedef for "base_metadata_type"
================================================================================
{
    "_type": "datatype",
    "name": "base_metadata_type",
    "description": "Definition of the base metadata type providing fields for
        the inclusion of outside links, submissions, and information pertinent
        to the experiment but not critical for the computation.",
    "_fields": {
        "name": {
            "_type": "string"
        },
        "creation_date": {
            "description": "The time and date that the object is
                created/computation was run",
            "_type": " datetime"
        },
        "modification_date": {
            "description": "This field will list all the dates and time an
                object ismodified.",
            "_type":" datetime",
            "_plural": "true"
        },
        "publication_status": {
            "description": "This is a choice field with four options. The draft
```

```
                    flag indicates that an object is in draft form and is still
                    being added to or changed. The open access flag indicates that
                    an object has been published and is freely available to anyone.
                    The objects with the private flag have restrictions on who can
                    view and access them. This is a way for researchers using
                    restricted data or metadata to insure the confidentially is
                    maintained.",
                "_type": "string",
                "_limit": {
                    "choice": ["draft", "open access",  "private", "recalled"]
                }
            },
            "authors": {
                "description": "This is the list of authors submitting this object.
                    First and last name are required, middle name and affiliation are
                    not. The affiliation field is for listing the academic,
                    governmental, or private entity that supported the project
                    responsible for generating the biocompute object.",
                "_plural": "true",
                "first_name": {
                    "_type":"string",
                    "_vital": "true"
                },
                "middle_name": {
                    "_type": "string"
                },
                "last_name": {
                    "_type": "string",
                    "_vital": "true"
                },
                "affiliation": {
                    "_type": "string"
                }
            },
            "keywords": {
                "description": "Keywords to aid in search-ability and description of
                    the object. These are a controlled vocabulary.",
                "_type": "string",
                "_plural": "true"
            }
        }
    }
}
```