The Information Flow Framework (IFF)

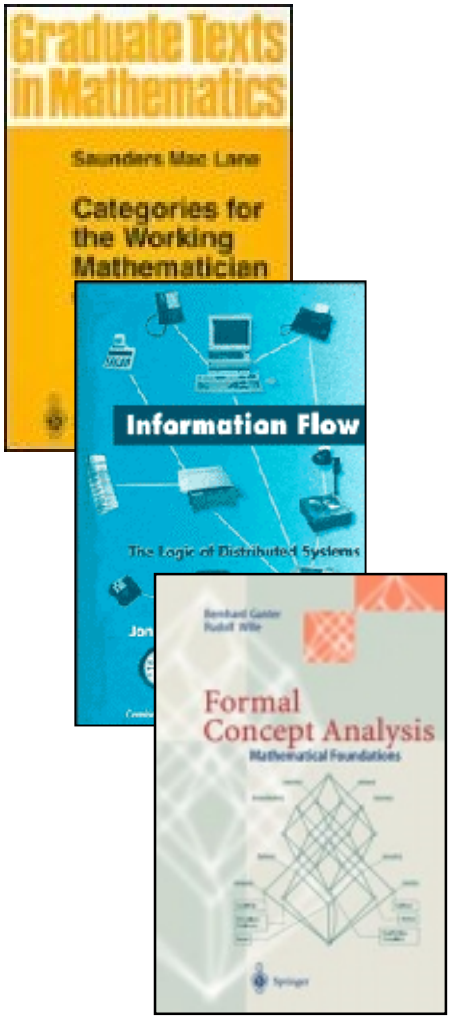
http://suo.ieee.org/IFF/

"*Philosophy cannot become scientifically healthy without an immense technical vocabulary. We can hardly imagine our great-grandsons turning over the leaves of this dictionary without amusement over the paucity of words with which their grandsires attempted to handle metaphysics and logic. Long before that day, it will have become indispensably requisite, too, that each of these terms should be confined to a single meaning, which, however broad, must be free from all vagueness. This will involve a revolution in terminology; for in its present condition a philosophical thought of any precision can seldom be expressed without lengthy explanations.*"

Collected Papers 8:169. Charles Sanders Peirce

What it is?

- consider the distinction between content and form
- this is basic in the general grammar of natural languages, in logic and in ontology
- the IFF offers a coherent principled approach to form
- this approach is realized in the structuring, mapping and integration of ontologies
- the IFF advocates a building blocks approach to ontology construction and management



Where it is being developed?

- under the auspices of the IEEE SUO WG
- it was the first starter document accepted by the SUO WG
- it aims to represent the structural aspect of the SUO
- the first version was submitted 20 July 2001 and accepted by vote 31 August 2001

What influenced it?

Information Flow

- the logic of distributed systems; a mathematically rigorous, philosophically sound foundation for a science of information.

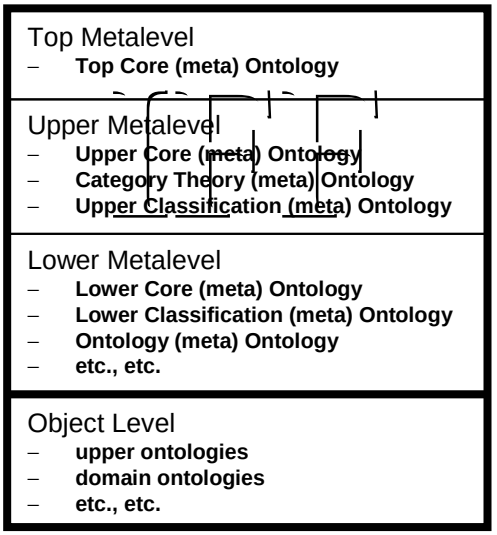
Formal Concept Analysis

- advocates methods and instruments of conceptual knowledge processing that support people in their rational thinking, judgments and actions.

Category Theory

- the study of structures and structure morphisms; starts with the observation that many properties of mathematical systems can be unified and simplified by a presentation with diagrams of arrows.

Created by Robert E. Kent



IFF Architecture

- The IFF architecture consists of three metalevels, numerous namespaces and (currently) seven meta-ontologies.
- Within each metalevel
  - the terminology is partitioned into namespaces
  - various namespaces are collected together into meaningful (possibly overlapping) composites called meta-ontologies.

3. Top Metalevel

- **Purpose:** provides an interface between the SUO KIF logical language and the IFF upper metalevel; services (represents and axiomatizes) the upper metalevel
- **Content:** the Top Core (aka Basic KIF) (meta) Ontology

2. Upper Metalevel

- **Purpose:** services (represents and axiomatizes) the lower metalevel; axiomatizes the structure of the lower metalevel via category theory; axiomatizes the IFF approach to the lattice of theories via formal concept analysis; organized in order to be able to realize the categorical principle[†].
- **Content:** partitioned into three (meta) ontologies – the Upper Core (meta) Ontology, the Category Theory (meta) Ontology, and the Upper Classification (meta) Ontology.

1. Lower Metalevel

- **Purpose:** services (represents and axiomatizes) the object level; location for various modules that help represent the structural aspect of the SUO using a categorical expression; used to the build the objects and morphisms of ordinary categories; should follow as closely as possible the categorical principle.
- **Content:** the Lower Core (meta) Ontology, the Lower Classification (meta) Ontology, the Ontology (meta) Ontology; possible future modules include, a module for "soft computation" (both rough sets and fuzzy logic), a module for theories of semiotics, a module for game-theoretic semantics, a module that corresponds to the Kestrel Institute's Specware system by representing the notions of sheaves and specifications, a module corresponding to the work by Goguen and Meseguer on institutions, etc.

0. Object Level

This is where the content ontologies reside, whether very generic, such as a 4D ontology, or more specific, such as ontologies for government or higher education; object-level ontologies satisfy a representation property similar to the categorical principle satisfied by the lower metalevel meta-ontologies – the ontological language used consists of the terminology defined and axiomatized in the lower metalevel.

[†] **Categorical Principle**: The design of any module situated in the IFF lower metalevel should be expressed at the level of set-theoretic classes and their functions, but using no quantification, no logical connectives, and only the specific terminology organized using composition, limits, and other categorical notions. This principle should make inferencing highly efficient.

Created by Robert E. Kent

IFF Language Analogy

- The IFF is coded in the SUO KIF logical language.
- The SUO KIF logical language is not the sine qua non for ontological expression.
- The IFF terminology is partitioned into language levels according to the following programming language analogy.

0. The KIF logical language = ontological machine language.

1. The Top Core (meta) Ontology terminology = ontological assembly language
   - an interface between the SUO KIF language and other ontological terminology
   - a "bootstrap" ontology

2-3. The upper/lower metalevel namespaces = a high level programming language (e.g. Lisp, Java, C++, ML, etc.)
   - encodes the bulk of the IFF
   i. The lower metalevel provides direct representation capabilities
   ii. The upper metalevel provides indirect structuring capabilities

4. The object level of the SUO = software applications (e.g. word processors, browsers, spreadsheet software, databases, etc.)
   - provides SUO content from the various domain ontologies
   - specified using the IFF lower metalevel terminology

- A complete distinction and an explicit boundary must be kept between the object level and the metalevel.

IFF Theory Structures

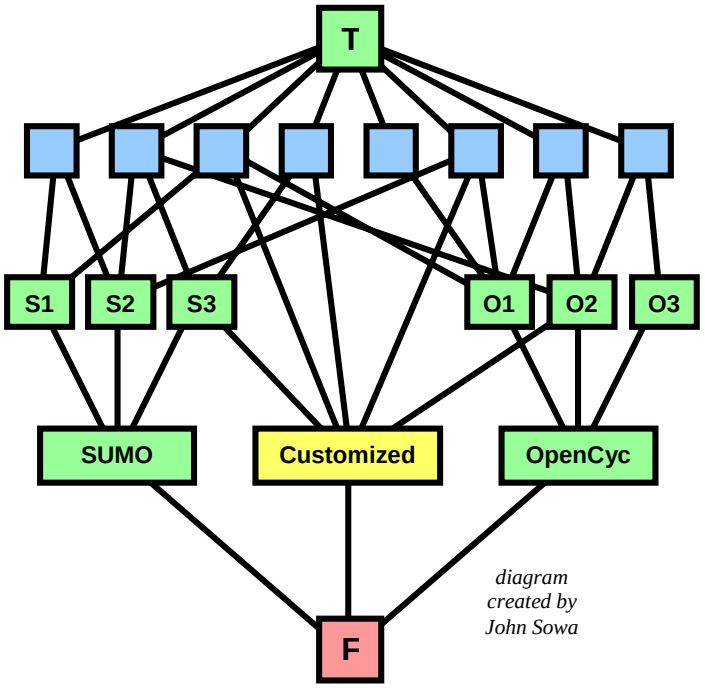


diagram created by John Sowa

The Truth Concept Lattice (TCL)

- The TCL consists of closed theories ordered by the opposite of subset inclusion.
- It is determined by the truth classification (models satisfying expressions).
- It is a true complete lattice with two (dual) sets of "generators" (expressions and models).

The Lattice of Theories (LOT)

- The LOT is determined by TCL and closure
  clo : theories $\rightarrow$ closed theories,
  where $clo(\boldsymbol{T})$ consists of all expressions entailed by $\boldsymbol{T}$.
  - $\boldsymbol{T}_1 \leq \boldsymbol{T}_2$ <u>when</u> $clo(\boldsymbol{T}_1) \supseteq clo(\boldsymbol{T}_2)$ <u>iff</u> $clo(\boldsymbol{T}_1) \supseteq \boldsymbol{T}_2$ <u>iff</u> $\boldsymbol{T}_1$ entails each $\boldsymbol{T}_2$ axiom
  - $\boldsymbol{T}_1 \equiv \boldsymbol{T}_2$ <u>when</u> $clo(\boldsymbol{T}_1) = clo(\boldsymbol{T}_2)$
- It approximates a true lattice; meets/joins unique up to equivalence; a complete pre-order with two (dual) sets of "generators" (expressions and models).

The Library of Modules

- Such a library is a navigable generalization/specialization hierarchical structure.
- There are actually two kinds of module libraries, one for TCL and one for LOT; a TCL library of modules is called a *truth collective concept* (in the IFF-UCLS).
- A library of modules represents
  - mutually consistent pairs of modules: a non-bottom module below them
  - overlapping pairs of modules: a non-top module above them
  - immediate generalizations and specializations (extended by iteration).

The Context (Category) of Theories (COT)

- The COT is indexed by the context of languages: a language underlies each theory.
- The COT is compatible with and partitions the LOT constructions.
- The COT is a proper extension of the LOT construction
  - COT theory morphisms restrict to LOT lattice ordering
  - COT sums[‡] correspond to LOT meets
  - COT quotients[§] have no LOT correspondents.

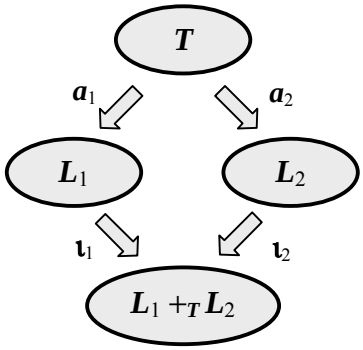
The Context (Category) of Closed Theories (CCT)

- The CCT is compatible with and partitions the TCL constructions.
- Etcetra.

[‡] When organizing a collection of ontologies in a lattice of theories, the underlying languages need to be summed by disjoint union.

[§] Similarly, terms regarded as equivalent need to be asserted equivalent and then quotiented.

Created by Robert E. Kent



IFF Semantic Integration

- The IFF has a principled approach to the semantic integration of object-level ontologies.
- Scenario: two communities want to design a fusion (merged) ontology
  - knowledge is represented in their own separate community ontologies
  - fusion ontology integrates the two participant community ontologies
  - fusion should respect the community knowledge structures
  - fusion should incorporate a substantial amount of agreement
- The IFF semantic integration accomplishes this
  - it works on object-level ontologies
  - it is the two-step process of alignment and unification.

Ontological Alignment

- the sharing of common terminology and semantics through a mediating ontology
- each community formalizes the common semantics via a local alignment link
- alignment is not an automatic process, but at best only semiautomatic

Ontological Unification

- results in a virtual fusion ontology of community connections
- fusion of the alignment diagram of participant community ontologies
- quotient of the sum of participant ontologies modulo alignment structure
- an automatic process

Representation of Ontologies

populated ontologies = those with instance data

- unpopulated ontologies represented by IFF theories
- populated ontologies represented by IFF logics

Integration Process

1. alignment
2. unification

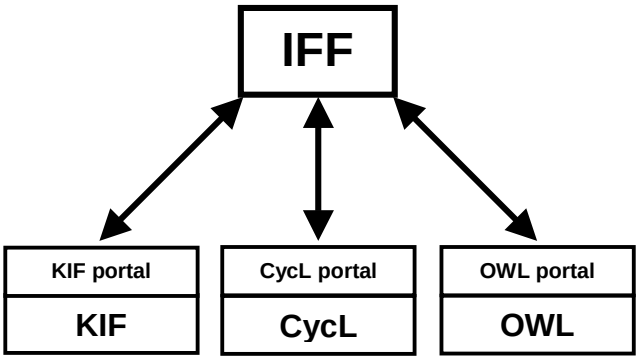
start

- participant community ontologies represented as logics $\boldsymbol{L}_1$ and $\boldsymbol{L}_2$
- integration facilitated by mediating or reference theory (ontology) $\boldsymbol{T}$
- alignment links $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ (theory morphisms) represent common semantics locally

finish

- completion of the alignment and unification steps
- resulting fusion logic $\boldsymbol{L}_1 +_{\boldsymbol{T}} \boldsymbol{L}_2$ represents fusion ontology
- embedding links ι_1 and ι_2 (logic morphisms) connect the original community logics with the fusion logic

Created by Robert E. Kent



The IFF Interface

- This interface gives the users view of the IFF.
- It serves as mechanism for ontology management, integration and interoperability.

Semantics is abstract

- Formal meaning is specified by the axioms; synonyms have the same meaning.
- Gedanken experiment
  - a copy $\boldsymbol{T}_2$ of a large theory $\boldsymbol{T}_1$ with just one term t_1 replaced by a second term t_2
  - then term t_2 has the same formal meaning as t_1.

Semantics can be complicated

- Repeated summations/quotients cannot be understood using a concrete representation.
- Gedanken experiment
  - 3 ontologies need to be organized into a lattice of theories and 25 pairs of terms are synonyms; later, 4 more ontologies need to be added to the same lattice of theories and 43 pairs of terms are synonyms; still later, 2 more ontologies need to be added to the same lattice of theories and 15 pairs of terms are synonyms
  - how do we manage the repeated summing and quotienting?
- motto: let the IFF abstraction work for you
  - input each ontology via portal from external representation into the IFF
  - manipulate the ontologies via portal, both yours and others
  - output each ontology via portal from the internal IFF representation.

IFF Portals

- An IFF portal = a point of access; it is a namespace that serves as an interface.
- It is used for communication between some external representation[**] (KIF, CycL, OWL, other XML, etc.) and the internal abstract IFF representation.

Input/Output Portal

- An IFF input/output portal is a mechanism for external/internal transformation
  - the external representation (terminology and axiomatization)
  - the internal abstract IFF representation of languages, theories and logics.
- It should be simple and automatic.

Control Portal

- An IFF control portal should have functionality for various internal processing
  - processing in the COT (summing, quotienting, etc.)
  - processing in the LOT (navigation, mutual consistency, meets and joins, etc.).

[**] An adequate external first order container/controller language will need augmented terminology for the various IFF classes, relations and operations – it should be able to express the IFF interface.

Table of Contents