Towards a Semantic Web of e-Governance: An Ontological Schema of the Constitution of India

Navjyoti Singh[1]*, Abhishek Sainani[1]

ABSTRACT

The constitution of India is not just a textual document; it brings into force an ontological environment that populates people and land with specially constituted and constrained entities/events. Entities such as citizen, legislator, tax payer, voter, company, district etc. are created and constrained by the constitution. We present a formal ontology to specify the constitution of India into a machine. This move allows aligning various e-governance initiatives into a unified machine-readable semantic framework. Various distributed databases of the government can thus become interoperable in the backdrop of this constitutional environment. Thus, search becomes plausible across the web ofe-governance projects.

Keywords: Ontology, e-Governance, Constitution, Point-Triple, Interoperability, Search.

1. Introduction

Information and communication technology is affecting processes of governance across the world. A large amount of digital data is being generated and retained by governments. Varieties of such databases, however, are not interoperable today, in spite of dealing with entities and events directly related to the requirements of governance. E-governance initiatives being undertaken by different agencies of a government cannot be easily related. Ontological engineering provides a new approach by which interoperability among databases and other government data can be achieved. A machine-readable ontology of government can make possible a unified schema for interpretation and semantic computing of a wide variety of such public interest data. We propose an ontology of government, using which a semantic web of e-governance can become a reality.

A standard approach to the problem of interoperability of government data is based on a taxonomy of relevant terms and unearthing class hierarchy among them (NZG LS, 2008) (AG IFT, 2008) (UK PGT, 2008) (GoC CST, 2008). Ontologically inclined approaches add few relations to supplement the taxonomy and hierarchy. However, the approach we are taking is altogether different from taxonomy-oriented ontology as we propose to deal not only with varied databases of the government but also unstructured data generated by the judiciary and parliament. A wide variety of data generated by various agencies of the state is relatable only through the constitution of the country. Thus we take a foundationalist and open approach to the ontology of government. Ontology of government can be considered as the system of its constitution. The constitution, codified as a written document, deals with specific entities, relations, events and rules governing them. The Indian constitution is such a document (CoI, 2008), which specifies in a principled

[1] Center for Exact Humanities, International Institute of Information Technology, Hyderabad 500 032, A.P., India
* *Corresponding Author:* (E-mail : singh.navjyoti@gmail.com, Telephone: +91 9951076012)

way the *ontological environment* of a unified Indian polity. We shall convert the structured document of the constitution of India into an ontological environment which is readable and interpretable by a machine.

Behind the codified document of Indian constitution is a huge graph/lattice of objects, rules and procedures. We call this lattice an 'ontological environment' of the constitution under the force of which the Indian polity works. We have evolved a method to specify this environment into a machine. The method uses Neo-Vaiśeṣika (Navjyoti, 2001) Formal Ontology NVFO and the idea of an ontological form called punctuator (Navjyoti, 2003). Punctuator is specified as a *point-triple* with the format 'object$_1$-RELATEDTO-object$_2$'. An ontological schema of the Indian constitution is codified as a set of point-triples in such a way that reasoning with them is decidable. Since databases related to governance involve constitutional objects like citizen and operative rules, databases are extractable as point-triples of interest. Further, such point-triples from databases can be interpreted in a machine-specified ontological environment. In short, data from these databases can be aligned well with the ontological environment and can be machine manipulated or evaluated. For instance, search across such data owned by different agencies can be done. Thus, the ground for semantic computing for the web of government data is created. This exercise on the Indian constitution can of course be replicated for other nation states.

The rest of the paper is organized as follows. In section 2, an ontology of constitutional objects is presented. We give a basic idea of an ontological form using which machine specification of ontology could be effectively done. Then we present a machine-readable ontological core of the constitutional environment. This is supplemented with schemas for reasoning in the constitutional environment. In section 3, we outline the actual encoding of the constitution of India using the framework of ontological core and associated reasoning schemas. In section 4, we deal with the decidability of the encoded ontological environment of the constitution of India. In section 5, we discuss the integration of e-governance initiatives with the ontological environment before we conclude.

2. Ontology of Constitutional Objects

The constitution of a country is not merely a document; it is an ontological environment that encompasses land and people who inhabit the land. Constitution brings into being hordes of objects that get possessed by the land and its people. Constitution brings into force relational entities that delimit and constrain the behavior of constituted objects. Further, constitution establishes rules that permit, prohibit, enjoin and constrain events in the polity. The aim of this section is to outline an ontology framework in such a way that the constitution of India becomes machine readable and decidable.

2.1 Machine Specification of Ontology

We use Neo-Vaiśeṣika Formal Ontology NVFO (Navjyoti, forthcoming) to specify the ontology of the constitution of India in a machine. The key idea of NVFO is an ontological form called *punctuator*. Punctuator is a form of contiguity between two entities in a relational context, where the relational context comprises of nested chains of punctuator-entity-punctuator. Thus, the form of punctuator is recursive because punctuators occur within the relational context of a punctuator. The logic of punctuator allows composition and decomposition between atomic relations and complex relations between two entities. It allows reasoning about a relational context as well as reasoning regarding mutual stitching of punctuators. We shall deploy these to specify the ontology of the Indian constitution to a machine and to automate semantic reasoning.

Punctuator can be simply specified as a triple "object-relation-object". Each such triple will be called a *point*. The ontology is specified in points using the format of triple – ⟨Object$_1$ RELATEDTO Object$_2$⟩. Machine reasoning will be done (1) by encoding composition/decomposition rules for middle terms in points, and (2) by encoding stitching rules for joining and forking of points. Unlike the form of proposition

"subject-predicate-object", punctuator formalism keeps middle term open for substitution, that is, middle terms can admit of granularity (specifiable at various levels of decomposition). This bestows sufficient freedom to seamlessly model varied domain ontologies in a way that semantic reasoners for domains take the burden of scalability and interoperability. These powers of punctuator formalism will get demonstrated below.

We have consciously not adopted a tree grammar based approach to specify constitution owing to the ontological richness of constitution. Thus we have not attempted XML and RDF style rendering of constitutional entities. The ontological richness of constitution can only be handled in more expressive graph grammars. Point-triples would allow graph grammar and graph theoretic computing. Point-triples are akin to *facts*. Our concern in machine specification is regarding constitutional facts and reasoning about them. Neither do we begin our ontology with class hierarchy. It is not that we shall not specify multiple class hierarchies, but we shall not render the ontology in terms of mono-hierarchy. The semantic computing style we adopt is close in spirit to two successful semantic web applications based on *fact* computing – (1) Powerset (2008) semantic search engine and (2) Yago fact handler (Suchanek, Kasneci & Weikum, 2007). It is thus understandable that we desist from rendering the ontology of constitution in the style of OWL. This in fact makes our specification scheme more expressive of the ontological environment of constitution and our reasoning more diverse and accommodative of real environment.

2.2 Objects and Points of Constitution: Framework of the Ontological Core

The ontological environment of constitution involves three kinds of objects – *Natural Objects*, *Constituted Objects* and *Fiat Objects*.

Natural objects are those that are independent of the constitution. For natural objects to be, it does not matter if constitution is promulgated or not. The two most important such objects are *HumanBeing* and *Land*. For, without these two objects, constituted objects simply cannot come into force. The third important natural object it requires is *Wealth,* which comprises of artifacts from productive activity of human beings, and entities like minerals, vegetation and animals. *Time* and *Space* are also natural objects. Apart from these five kinds of natural objects, which play a major role in delimiting the very being of constituted objects, the constitution could refer to other specific natural objects as well in a specific context. Constituted objects are of two kinds – *Institutions* and *Offices*. These objects are brought into being by the constitution. Institutions are like state, parliament, company, trust etc. and offices are like president, speaker of parliament, chief justice, election commissioner, citizen etc. It is important to think of citizen as a kind of office that is fundamentally different from human being, which is a natural object.

Fiat objects are divided into three major kinds – *Territory*, *People* and *Culture*. These can be characterized using point-free geometry of region and boundary. Territories are fiat boundaries of land and constitution invokes several of them like territory of country, states, cities, villages, land holdings etc. People are 'conditioned collectivity' of human beings, like people of India, weaker sections in society, religious minority, minors, women, belonging to sub-region etc. Culture is a feature possessed by people, like nation or national culture, regional culture, legacy culture, religions etc. Culture is delimited by traits like language, manners, dress, arts, food etc.

These objects can be specified in a machine as sets of points in the manner given below:
Object ISOFKIND NaturalObject
Object ISOFKIND ConstitutedObject
Object ISOFKIND FiatObject

NaturalObject ISOFKIND HumanBeing
NaturalObject ISOFKIND Land

NaturalObject ISOFKIND Wealth
NaturalObject ISOFKIND Time
NaturalObject ISOFKIND Space

ConstitutedObject ISOFKIND Institution
ConstitutedObject ISOFKIND Office

Institution ISCONTAINEROF Institution
Institution ISCONTAINEROF Office
Institution LARGEST State
Office SUBORDINATETO Office
Office SMALLEST Citizen
Office BIGGEST President

FiatObject ISOFKIND Territory
FiatObject ISOFKIND People
FiatObject ISOFKIND Culture

Land ISSPATIALFIATBOUNDIN Territory
HumanBeing ISCOLLECTIVEFIATBOUNDIN People

People POSSESS Culture
Culture DELIMITEDBY Trait
People HASSECTION People

Mutual dependency of natural and constituted objects is based on the *doctrine of limited sovereignty*. Human being is the only natural sovereign object in terms of having the autonomous natural capacity to decide and act judiciously. However, human being partially surrenders this capacity for the sake of brevity in living. They partially leave decisions regarding their activities to others. This suspension of the sovereign role and capacity accumulates in State and offices contained in State. The power issuing out of such limited sovereignty is devolved into various offices of State. Logic behind the motion of sovereignty and construction of authority is given in the second chapter of Navjyoti (forthcoming). Dynamics of sovereignty is a fundamental phenomenon which any constitution handles. We can specify it in the way given below:
HumanBeing CONDUCTS Activity
HumanBeing INHERE Sovereignty
HumanBeing PARTIALLYTRANSFER Sovereignty
Sovereignty ACCUMULATESIN State
Sovereignty DISCHARGEDBY President

Sovereignty BESTOWS Power
Sovereignty BESTOWS Function
Sovereignty BESTOWS Privilege
Sovereignty BESTOWS Responsibility

President DEVOLVES Power
President DEVOLVES Function
President DEVOLVES Privilege
President DEVOLVES Responsibility

Office ISOFKIND DevolvedOffice

DevolvedOffice HASATTRIBUTE Power
DevolvedOffice HASATTRIBUTE Function
DevolvedOffice HASATTRIBUTE Privilege
DevolvedOffice HASATTRIBUTE Responsibility

HumanBeing OCCUPY DevolvedOffice
HumanBeing INHERE Citizen

Institution HASATTRIBUTE Power
Institution HASATTRIBUTE Function

Citizen HASATTRIBUTE Right
Citizen HASATTRIBUTE Duty

Another important feature of the constitution is the establishment of institutional structures such that checks and balances in the discharge of devolved power can be hardwired. The *Doctrine of tripartite division of power* into executive, legislative and judicial wings is a result of such concern. The President primarily devolves power in accordance with this institutional division. This can simply be specified as the following declarative point-triples:

State ISA Institution
Legislature ISA_Institution
Executive ISA Institution
Judiciary ISA Institution
Consolidated Fund of State ISA Institution
…….List of Institutions

President ISA Office
Auditor General ISA Office
……List of Offices

Further, interdependence in the functionality of offices and institutions is also hardwired in the ontological environment of the constitution. Constitutional offices have to inform other offices regarding events in their purview. Offices can also order other offices for enforcement of certain functions. This way, coherence in the discharge of power is achieved. For the functionality of office to be possible, occupation of office by human being is required. The details of interdependence with respect to any two offices can be specified as:

Office INFORMS Office
Office DIRECTS Office
Office ADVISE Office
Office ADVISE HumanBeing
HumanBeing INFORM Office

Activities of all objects and institutions in their discharge of power and functionality as well as activities of human beings in the purview of State are governed by law. Laws are a set of rules that permit, prohibit or restrict activities of authorities and human beings. These rules are enforced by institutions. Depending on the domain of activities, there are different sets of laws like constitutional law, administrative law, criminal law, civil law, trust law etc. All of them can be specified in the following general framework:

Institution ENFORCE Rules
Rules PERMIT activity
Rules PROHIBIT activity
Rules RESTRICT activity

Constitution requires for its enforcement registration of a variety of objects definable in the above schema. For example, the object 'citizen' comes into force when it inheres in an actual human being. This requires registering the actual date of birth of a particular human being. Registration is a mechanism of specifying the link between a constituted object and a particular natural object. Only when particulars are registered can a constitution in reality come into force. Promulgation alone does not operationalize constitution. It is for this reason that constitution provides for a variety of registration and certification offices. These can be specified as:

Office ISOFKIND Registry
Birth_Registrar ISA Registry
Birth_Registrar HASFUNCTIONTOREGISTER Birth
Voter_Registrar ISA Registry
Voter_Registrar HASFUNCTIONTOREGISTER Voter
……List of Registries

In a nutshell, the above given ontological environment is a core framework of constitution, using which all objects and directions mentioned in the constitution can be specified in a machine. The Constitution of India, for instance, can be specified in detail in terms of filling in the details of this environment. In fact, this generalized environment has been arrived at after detailed analysis of the framework of the constitution of India. Even then the schema itself is general enough to be able to present constitutions of other countries after suitable minor revisions. The framework is general enough to withstand a claim that construction of the actual constitution is filling up of the framework. For instance, details of the lists of institutions, lists of offices, lists of registries along with their attributes, interdependence and norms are encoded as the document of the constitution. The exact devolution of power to further construct or register more institutions and offices is also specified in the actual constitution.

2.3 Schemas of Middle Terms in Point-Triples: Reasoning in the Ontological Environment

After specification of the ontological framework of the constitution is achieved, a machine should be able to reason about middle terms in the point-triples. To handle middle terms we need the following reasoning algorithms:

- Schema for Universals and Particulars: Logic of ISOFKIND & ISA is based on a strict hierarchy of universals inhering in terminal instances. Closure of this domain is achieved by positing a highest universal at the top and terminal particulars at the bottom.
- Schema for Container and Contained: Logic of ISCONTAINEROF & SUBORDINATETO is based on the exclusivity of container. Closure in this domain is achieved by positing the largest container which is State, the biggest office of President and the smallest office of Citizen.
- Schema for Bounded and Bounds: Logic of ISSPATIALFIATBOUNDIN & ISCOLLECTIVEFIATBOUNDIN is based on point-free geometry of discrete regions (Clarke, 1981) (Clarke, 1985). Since 'bounded' is an object, say, land or human being, decidability rides on actual objects being bound and calculus of regions. Positing the largest spatial region and largest collective region brings closure in this domain. Two human beings form the smallest collective region and material atom forms the smallest spatial region. Logic of HASSECTION governs non-exclusive inclusions dealing with collective regions which are conditionally bound.

- Schema for Attributed and Attribute: Logic of HASATTRIBUTE also rides on the underlined logic of objects which are attributed by it. However, shared attributes among objects can be analyzed in terms of universals residing in them like that of kinds of responsibilities or duties. Attributes are handled in a directed graph theoretic way as connected nodes (Navjyoti, forthcoming).
- Schema for Human Activities: Logic of INHERENCE is based on the NVFO ontology, wherein the human individual is a structured compound of three ultimate substances – matter, *manas* and self. Individual is unified by a mechanism that bestows unified action to a compound. Logic of the motion of sovereignty, that is PARTIALLYTRANSFER, is built on the base of such formal ontology of individual (Navjyoti, forthcoming). NVFO distinguishes compounded individual from 'person'. 'Person' inheres in the self and lacks body and motor organs. Even 'citizen' inheres in self and lacks body and motor organs. Person, as a natural object, dwells in the fiat container 'culture', whereas citizen, as a constituted object, dwells in the container 'State'.
- Schema for other Relations: CONDUCT, POSSESS, BESTOWS, DEVOLVES, OCCUPY, INFORMS, DIRECTS, ADVISE & ENFORCE are handled in a graph theoretic way as punctuators such as conducter|conducted, possessor|possessed, bestower|bestowed, devolver|devolved, occupier|occupied, informer|informed, director|directed, advisor|advised and enforcer|enforced. Even the DELIMITEDBY, ACCUMULATESIN, DISCHARGEDBY & HASFUNCTIONTOREGISTER relations are handled using general punctuators such as delimitor|delimited, accumulator|accumulated, discharger|discharged and registeree|registered. These are general forms different from each other in terms of their particular relational contexts. Analysis of relational contexts of these punctuators can be done when and where entailed or required. Such an analysis can be embedded in nodes of the edge of punctuator rendered in a directed graph theoretic way. For these relations we leave open the scope for 'granularity of analyses'.
- Schema for Rules: Logic of PERMIT, PROHIBIT & RESTRICT can be modeled by deontic logic of possibility and necessity (Hilpinen, 2001). Permitted actions are possibly entailed and prohibited actions are necessarily entailed. Restrictions can be seen as constraint rules specified for permission and prohibition of relevant actions of different objects.
- The ontological environment of the constitution of India can now be specified in a machine which is capable of semantic reasoning. The schema presented in this section is decidable as there is closure of all middle terms and entities are finite.

3. Converting the Constitution of India into Point-Triples

The Constitution of India is an extension of the core ontological environment and reasoning schema that has been sketched in section 2.

We have attempted to turn the Indian constitution into non-ambiguously specifiable point-triples. Lists of institutions and offices contained in them were made. Attributes, interactions, norms etc were recorded. In fact we converted the textual form of constitution as point-triples. While doing so, sometimes a new middle term has to be introduced in a particular context of the constitution. For instance, while handling the election of the President, we had to introduce ISELECTEDBY, ELIGIBILITY & HASVALUE. For them logic for reasoning had to be worked out. We also include *id* for each point-triple for locating that triple in the long list of point-triples so that chains and embedding of point-triples into one another can be done.

We give an example of the election of the president as we encountered the need to introduce new middle terms when we reached it. We also had to introduce reification while encoding it, where the occupier|occupied punctuator was turned into an object for another point-triple. From Article 53 to Article 59 of Indian Constitution deal with the office of Presidency. We give below the point-triples of the relevant section just to illustrate the kind of difficulty faced while encoding the constitution:
President ISELECTEDBY PresidentialConstituency

PresidentialConstituency ISOFKIND HouseofPeople
PresidentialConstituency ISOFKIND CouncilofStates
PresidentialConstituency ISOFKIND LegislativeAssemblyofState
HouseofPeople ISCONTAINEROF HPMembers
CouncilofStates ISCONTAINEROF CSMembers
LegislativeAssembliesofState ISCONTAINEROF LAMembers
HPMember HAVEATTRIBUTE HPPresidentialVote
CSMember HAVEATTRIBUTE CSPresidentialVote
LAMember HAVEATTRIBUTE LAPresidentialVote
HPPresidentialVote HAVEVALUE *x*
(*x*= The Total Number of votes assigned to Members of the Legislative Assemblies of the States Divided by the Total Number of the elected Members of both Houses of Parliament)
CSPresidentialVote HAVEVALUE *y*
(*y* = The Total Number of votes assigned to Members of the Legislative Assemblies of the States Divided by the Total Number of the elected Members of both Houses of Parliament)
LAPresidentialVote HAVEVALUE *z*
(*z* = Multiples of one thousand in the quotient obtained by the Population of the State Divided by the Total Number of the elected Members of the Legislative Assembly)
Population OBTAINEDFROM LastCensus

Human Being OCCUPY President :#id
Occupation ISA Occupier|OccupiedPunctuator
Occupation ISOFKIND PresidentialOccupation
PresidentialOccupation ISA #id
PresidentialOccupation ISFOR Five Years
PresidentialOccupation HASSTARTCONDITION Election
PresidentialOccupation HASENDCONDITION Resignation
PresidentialOccupation HASENDCONDITION Impeachment
PresidentialOccupation HASENDCONDITION ExpirationofTerm

PresidentialCandidate ISA Citizen
PresidentialCandidate ELIGIBILITY Age (=more than 35 years)
PresidentialCandidate ELIGIBILITY HouseofPeopleCandidate
Presidential Candidate ELIGIBILITY NotHoldingAnyOfficeofProfit

In this example many terms in the middle or in the nodes are new and not specified in the ontological core. A schema for each such term has to be provided for enabling semantic computing.

4. Decidability of the Ontological Environment: Implementing the Stitching Algorithm for Point-Triples

Each object defined in the constitution is connected with every other object in some way or the other. Hence, the constitution of India, with entities as nodes and relations as edges, is a fully connected graph structure, where no object is left in isolation and each object is connected with some other object in the constitution. Given any two entities in the ontological environment, at least one path can be constructed between them by stitching point-triples. This precisely is the idea of the recursive form of punctuator.

The above encoded ontological environment is a set of atomic punctuators, each marking contiguity between two entities. However, any two entities in the environment can be contiguous provided there exists at least one path between them as their relational context. A path is obtained by linking atomic punctuators

on the basis of occurrence of same object in them. In fact, we shall get many paths between any two arbitrary entities. Thus a new punctuator can be constructed from a given ontological environment in a machine. The system is decidable if there exists a path between any two objects in the ontological environment in the machine.

Decidability is proved if (1) there is no object in the system which is not connected to some other object, and (2) there is no punctuator (or point-triple) in the system which is not connected to some other object in the system.

We define a code structure to stitch triples of constitution into paths between two arbitrary constitutional objects/instances. The idea is to take two objects as input. One would be the source, the other would be the destination. We take the left side object of the middle term or relational context, check for the right side object to see if it is the other object. If yes, then output the list of IDs corresponding to the triples which were traversed. If not, then find a left side object matching with the right side object of the current triple.

```
Class Punctuator
{
      public:
            String LeftObject;
 String Relation;
            String RightObject;
                              String ID;
            Punctuator *Next;

            Punctuator ()
            {
                  Next=NULL;
            }
};
```

The class Punctuator defines the basic structure of the point-triple. The pointer *Next would point to the next point-triple in the path to reach the other object. The other object would always be the right side object of the destination point-triple and the first object which starts the search would always be the left side object of the source point-triple.

Approach for the path finding function:

- Find an instance of point-triple, say sourceTriple, having source entity as LeftObject.
- Find immediate neighbors of sourceTriple, i.e., other triples having LeftObject same as RightObject of sourceTriple.
- Then treating the selected immediate neighbor as sourceTriple, find its immediate neighbors.
- Each immediate neighbor selected is joined to the list of triples that make the path from the source entity to the destination entity.
- We continue step 2 to step 4 for all the neighbors of each triple in the path, one at a time, till one of the neighbors having RightObject same as the destination entity is found. The corresponding path, or list of triples, is displayed as the path between the two entities. The algorithm terminates.
- If no path is found between the two entities, Steps 1 to 4 are repeated for some other triple having the source entity as LeftObject.
- The algorithm terminates formation of a path if it finds atomic entities as RightObject in a neighbor triple.

Approach for finding non-intersecting paths:

To find non-intersecting paths between the two entities, we make a few changes in the previous approach.

- As soon as one path is found between the two entities, the algorithm is not terminated, but executed again. This time, the triples contained in the path found by the previous execution of the algorithm are not considered as neighbors in the path formation process.
- For each time the algorithm is executed as in step 1, we get a list of paths; each having its own set of triples, and no triple is common among any of the paths found. Hence no two paths would be intersecting.

5. Integration of the Ontological Environment with e-Governance Initiatives

A Constitutional Environment exists in the governance and functioning of a nation. We have specified a decidable model of the constitution of India, which makes the constitution or constitutional environment machine readable and understandable. Various categories of objects specified have actual instances which are the subject matter of various e-governance databases. That is why the machine readable environment can be conveniently integrated with various e-Governance initiatives ((OSNeGP, 2008) which involve management of large databases like electoral database, census database etc. Schematic diagram of such an apex level integration is sketched in Figure 1. The integration becomes convenient simply because these government databases contain objects and attributes that can be specified as a part of the constitutional environment.

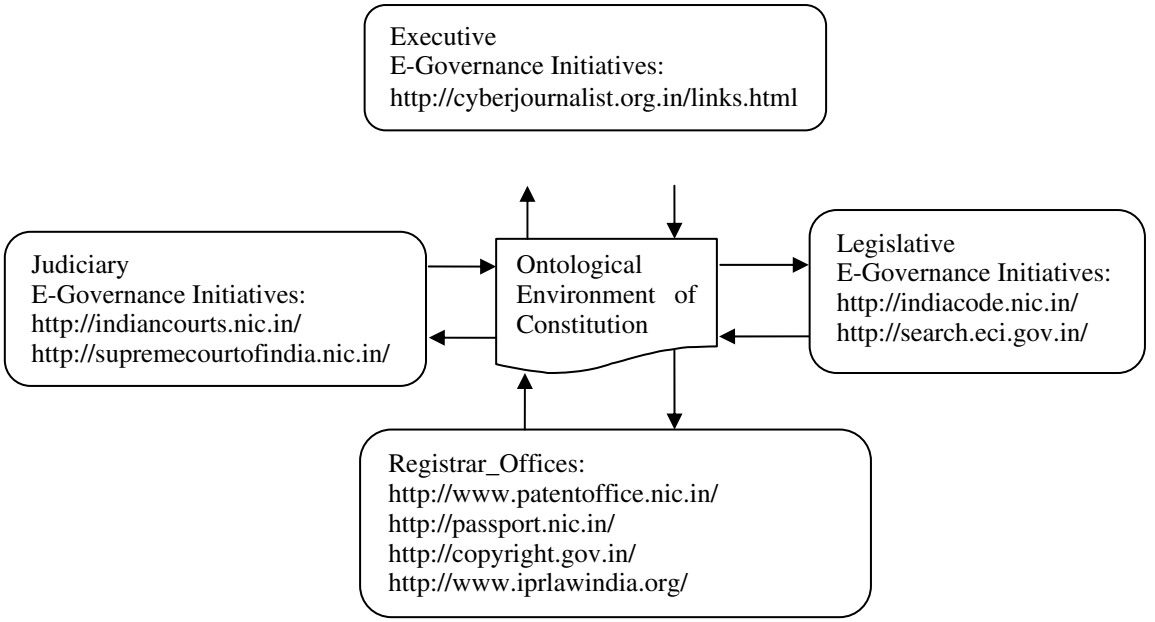

Figure 1: Various e-Governance Initiatives are linked to Ontological Environment of the Constitution

Let us take an example of an important component of an e-Governance initiative like registration. According to the constitution, various kinds of registration like birth registration or citizenship registration, property registration, license registration etc. should be done. Registration is necessary because constitutional categories have to be linked with actual natural entities like human beings and land. Registration is important and is done to maintain information about the state of natural entities like human beings within the constitutional environment. Each of these registries is a huge government database. These databases contain objects which are either objects defined by the constitution or objects that could easily be linked to constitutional objects, like their attributes. So, for a driving license, the candidate's age has to be

18 yrs or above. Age is not really a constituted object but is an attribute of the constitutional object called citizen because citizen inheres in a particular human being and in which inheres the attribute age. Human being and citizen both belong to the ontological environment as natural object and as constituted object respectively. Thus, a license registration database can be interpreted in the machine readable environment.
Another important example of the e-Governance initiative is e-documentation of various Acts of Parliament (ICIS, 2008) ((ECI, 2008). Acts of Parliament are usually directed at prescribing the law to be followed by the public or a class of persons or by the government in its dealings with others, either the public at large or a section of it. These Acts of Parliament are stored in government databases. They have to be translated and transformed into point-triple form according to the constitutional schema, which would make them interpretable by machine. Other huge data like judgments of courts (ICJD, 2008) (SCI, 2008) also have to be interpreted in point-triples compatible with the above specified environment. Similarly, orders of the executive (L-EO, 2008), which are issues on the daily basis, have to be interpreted in point-triples compatible with the environment. Point-triple schema of the constitutional environment is powerful enough in enabling interoperability not only among numerous structured databases but also among unstructured data like Acts of Legislature, Orders of Executive and Judgments of Judiciary.

Generally the different e-governance databases use incompatible schemas, which may be a problem in searching across the databases, or recognizing same entities in two or more different databases. So, to connect these databases and ensure interoperability among them, the e-governance databases are converted in a machine readable/understandable form (i.e., point-triples) and integrated with the point-triples depository of the constitution. The point-triples form of constitution thus bridges the syntactical differences among these databases. This makes searching across different public government databases feasible.

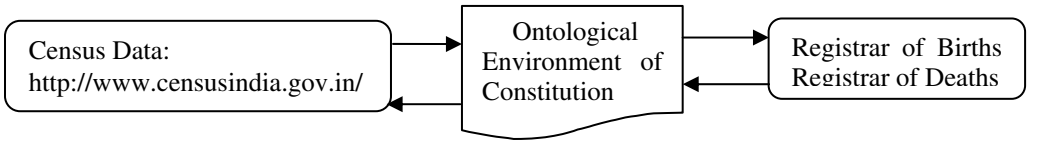

Figure 2: Interoperability between Census and Registrar Data

Interoperability among various government databases is important and useful because when a change corresponding to an entity occurs in one database, it should also be reflected in other databases concerning that entity. For example, in the case of birth and death registrations, the government of India has taken a lot of initiatives to ensure proper and timely registration of births and deaths by the hospitals or head of locality. But such and similar initiatives have led to nothing more than improvement in collecting and storing data for individual databases/purposes. It means that if the birth registration database for a region is immediately updated, no simultaneous change is reflected in the related databases which may concern the born child, but only when the need is felt. Likewise, the registration of death doesn't reflect in the records of that person in other government databases like tax payments or voter's database. The dead person is no longer a citizen of India, but since the databases/records are not simultaneously updated, his citizenship remains valid for a much longer time, which might lead to wrong statistics as well. In *Figure 2* is sketched natural interoperability path between census authority and registration of birth/death authority. Thus, every database where the record of the deceased person exists should to be updated separately.

Various e-government interoperability initiatives have been taken by some governments like Malaysia (MyGIF, 2003), New Zealand (NZ e-GIF, 2008), Asia-Pacific countries (GAPC, 2008) etc. Primarily either the Government Interoperability Framework GIF (Stevenson, Stevenson & Hill, 2005) (Lallana, 2008) or National Enterprise Architecture NEA (Lallana, 2008) methodology had been adopted for attaining interoperability in the e-government context during the initial phases. While GIF includes technical standards on interconnection, data integration, information access and presentation, metadata, and security,

NEA includes, apart from technical dimension, standards in organization/process and data/semantic dimensions as well. Interestingly most of the countries following GIF, while reviewing their interoperability standards, also included some standards in organization and data semantic dimensions as well.

Our approach focuses primarily on the non-technical part, i.e., semantics of data and organization of data, since the data and the structure of the data stored are all connected directly or indirectly to the constitution. We use the point-triples concept to represent the data (from databases or websites) and the point-triples form of constitution and the grammar of ontology of constitution to provide an environment of interoperability among various forms of data in e-governance.

6. Concluding Remarks

The novel idea of presenting the constitution as an ontological environment brings to light the possibility of a machine being able to interpret the structure and meaning of constitutional polity. This is indeed achieved because of our approach to formal ontology. Our approach depends on the idea of an ontological form – punctuator – rendered as point-triple. It is the recursive structure of punctuator which endows decidability to the proposed ontology of the constitution of India. With constitution of India rendered in point-triples, semantic reasoning by machine becomes possible. Not only can rationality of middle terms in constitutional point-triples be specified to machine but also nodes of point-triples can be stitched into path and reification of point-triples can be done. With constitutional environment specified in machine, we can conveniently integrate various e-Governance initiatives within this encoded ontological environment as government activities are ontological products of the constitution itself. This can be done by extracting various structured and unstructured data from government domains in the form of point-triples and aliening them with the environment. This would eventually lead to automation of the semantics of various ontological objects and attributes that occur in the digital web of the government. Interoperability of diverse government data can thus be achieved through underlying machine specified constitutional environment.
In future, we shall integrate Acts of Legislature and Orders of Executive and Judgments of Judiciary into the proposed ontological environment. We plan to develop a Semantic Wiki for the purpose, which would be both human-readable and machine-readable and which would provide an integration platform for various e-Governance initiatives. Semantic searching across government data would be possible as well. Semantic web of governance would thus be a reality.

References

1. (AG IFT) The Australian Governments' Interactive Functions Thesaurus. Available at: http://www.naa gov.au/recordkeeping/gov_online/agift/summary.html. Accessed October 24, 2008.
2. Clarke, Bowman (1981). A calculus of individuals based on 'connection'. Notre Dame *Journal of Formal Logic,* 22, 204-18 (1981).
3. Clarke, Bowman (1985). Individuals and Points. Notre Dame *Journal of Formal Logic*, 26, 61-75.
4. (CoI) Constitution of India. Available at: http://indiacode.nic.in/coiweb/welcome.html. Accessed October 24, 2008.
5. (ECI) Election Commission of India Site (ECI). http://search.eci.gov.in/. Accessed October 24, 2008.
6. (GAPC) Ensuring e-Government Interoperability: Guidelines for Asia-Pacific Countries (2008). Available at: http://ec.europa.eu/idabc/servlets/Doc?id=31218. Accessed October 24, 2008.
7. (GoC CST) Government of Canada Core Subject Thesaurus. Available at: http://en.thesaurus.gc.ca/intro_e.html. Accessed October 24, 2008.
8. Hilpinen, Risto (2001). Deontic Logic. In Goble, Lou, ed., *The Blackwell Guide to Philosophical Logic.* Blackwell.
9. (ICIS) The India Code Information System (containing all Central Acts of Parliament right from 1836 onwards). Available at: http://indiacode.nic.in/, http://india.gov.in/govt/acts.php/. Accessed October 24, 2008.
10. (ICJD) Indian Court Judgment Data. Available at: http://indiancourts.nic.in/. Accessed October 24, 2008.

11. Lallana, E. C. (2008). *e-Government Interoperability*. Available at: http://www.apdip.net/publications/iespprimers/eprimer-gif.pdf. Accessed October 24, 2008.
12. (L-EO) Links Related to Executive Offices and Links related to Government of India. Available at: http://cyberjournalist.org.in/links.html. Accessed October 24, 2008.
13. (MyGIF) Standards, Policies and Guidelines - Malaysian Government Interoperability Framework (2003). Available at: http://www.mampu.gov.my/mampu/pdf/ISPlan/ispdoc/Interoperability%20Framework.pdf. Accessed October 24, 2008.
14. Navjyoti Singh (2001). Comprehensive Schema of Entities: Vaiśeṣika Category System. *Science Philosophy Interface,* 5, (2), 1-54.
15. Navjyoti Singh (2003). Theory of Experiential Contiguum. In Philosophy and Science: Exploratory Approach to Consciousness (111-159). Kolkata: Ramakrishna Mission Institute of Culture.
16. Navjyoti Singh (forthcoming). Punctuating Reality: Towards Formal Foundations of Justice, History and Society. Shimla: Indian Institute of Advanced Study.
17. (NZ e-GIF) New Zealand E-government Interoperability Framework (NZ e-GIF) (2008). Available at: http://www.e.govt.nz/standards/e-gif/e-gif-v-3-3/e-gif-v-3-3-complete.pdf. Accessed October 24, 2008.
18. (NZ GLS)The New Zealand Government Locator Service Thesaurus. Available at: http://www.e-government.govt.nz/docs/interim-thesaurus/index.html. Accessed October 24, 2008.
19. (OSNeGP) Online Services under National e-Governance Plan. Available at: http://india.gov.in/govt/national_egov_plan_online.php. Accessed October 24, 2008.
20. Powerset. Available at: http://www.powerset.com/. Accessed October 24, 2008.
21. Stevenson, A., Stevenson, J., Hill, A. (2005). *e-Government Interoperability Framework*. Available at: http://www.archiveshub.ac.uk/arch/eGIF_march05.ppt. Accessed October 24, 2008.
22. Suchanek, F. M., Kasneci, G., Weikum, G. (2007). *Yago, a core of semantic knowledge*. In WWW 2007 (697-706).
23. (SCI) Supreme Court of India. Available at: http://supremecourtofindia.nic.in/. Accessed October 24, 2008.
24. (UK PGT) The UK Pan-Government Thesaurus. Available at: http://www.govtalk.gov.uk/documents/UK%20Metadata%20Framework%20v1%202001- 05.pdf. Accessed October 24, 2008.

About the Authors

Navjyoti Singh is a Professor and Head of the Center for Exact Humanities at International Institute of Information Technology, Hyderabad, India. After engineering education from Indian Institute of Technology Kanpur in Mechanical Engineering and Nuclear Technology, he shifted to professional research in philosophy and history of science. His research activity has been focused on theoretical inquiry at the crossroads of Humanities, Science and Indian Analytic Traditions, His publications and teaching are in the area of formal ontology, ontological engineering, theory of mind, society and justice, foundations of logic and mathematics and cultural computing,

Abhishek Sainani is a fourth year student of BTech-MS dual degree in computer science at International Institute of Information Technology, Hyderabad, India. He is interested in data mining from structured and unstructured public data. He has done advanced courses in data engineering, ontological engineering and theory of society, history and justice.