Chapter 4

An Ontology Based Modeling Framework for Instructional Design

The fundamental idea of domain knowledge and what kind of domain knowledge is required for adult literacy is discussed at length in Section §4.1. We then briefly discuss the broad spectrum of ontologies and their development process in Section §4.2 and Section §4.3 respectively. We focus on the breadth of ontologies in the domain of instructional design in Section §4.4. In Section §4.5, we present our ontology based framework for modeling instructional design. Within this section, we detail ontologies for modeling goals, instructional process and instructional material in sub sections §4.5.1, §4.5.2 and §4.5.3. We end the chapter by concretely presenting a domain ontology for adult literacy in Section §4.6.

4.1 Domain Knowledge

John McCarthy has envisioned that an intelligent way of building systems should focus on the knowledge that is required to represent system’s inputs and methods through which possible conclusions can be automatically derived from that knowledge [198]. Newell has proposed the need to have a knowledge level focusing on specifying the world independent of symbol level that focuses on implementing the behaviour of the system [199].

“The Knowledge Principle: A system exhibits intelligent understanding and action at a high level of competence primarily because of the specific knowledge that it can bring to bear: the concepts, facts, representations, methods, models, metaphors, and heuristics about its domain of endeavor.” -Lenat and Feigenbaum [200]

The ContentPattern proposed in Chapter §3; consisting of facts, cases, rules, models and theories can be considered as a part of domain knowledge in the above statement. Feigen-
Figure 4.1 Top-down decomposition of sentences to syllables [A], Bottom-up composition of words and sentences from syllables [B], Top-down decomposition of sentences to phonemes [C]
baum coined the term *knowledge engineering* and proposed knowledge base should be a fundamental basis that stores expertise of human experts in solving real-world problems [200]. There are two primary directions of research related to knowledge engineering one in the field of Artificial Intelligence primarily for automatic reasoning and expert systems and in computer science to represent different aspects of the system. Several researchers have figured out multiple ways of representing knowledge like concept maps, topic maps, ontologies, first order logic and so on [201][202].

What is the specific domain knowledge in this thesis and how to represent it? In order to address this question, we start with concrete examples from adult literacy case study setting the context for the rest of the chapter. We introduced IPCL in Section §2.1 as a well-established approach for teaching adult illiterates in India. This approach suggests the use of eclectic method for teaching reading skills, comprehension, problem solving and facilitates learning through interpretation of contents in the context of life [76]. We consider “context of life” as mentioned in IPCL approach as learners’ prior knowledge. Eclectic method primarily differs from traditional methods as it does not start with alphabets but instead uses familiar and known words to learners, decomposes them to syllables and phonemes as shown in Figure §4.1[A][C].

These syllables and phonemes are further synthesized to form words and in the end, the alphabet is learnt as in Figure §4.1[B]. The patterns of decomposition (top-down) focuses on cognitive abilities of learners whereas the patterns of composition (bottom-up) can facilitate reasoning of the subject knowledge. We have depicted examples of this process in Figure §4.2 for the Telugu language based on the primer (instructional material) and IPCL approach. In this figure, a sentence in Telugu language కలం అన్గం is first decomposed into two words namely కలం and అన్గం. Each of these words are further decomposed till the syllables are obtained. Similarly, the same sentence is also decomposed into phonemes representing the sounds of the sentence, words and syllables respectively. On the other hand, Figure §4.2[A] shows the composition process which uses the syllables and phonemes to form words and sentences. Specifically, words కల, ఊక and అందం are formed by composing the syllable క and other relevant syllables and words అందం, వన are formed from their respective syllables with the learnt syllable అ in red color. This hierarchy of decompositions and compositions forms the basis for learning new syllables and phonemes in a language as shown in Figure §4.2[B]. Several compositions are possible V-Vowel+Vowel Modifier, C-Consonant+Vowel Modifier, C+V-Consonant+Vowel, Vowel Modifier, C+C-Consonant+Consonant, C+C+V-Consonant+Consonant+Vowel and there will be several constraints on these compositions as well. For example, only one modifier might be allowed after a consonant. Discussing that in detail is not within the purview of this thesis. But this approach of learning alphabet from known words and sentences and later synthesizing new words from syllables and phonemes was empirically established by IPCL as a successful approach for adult literacy in India [26][203]. Most importantly, this approach works
Figure 4.2 Composition of cases from facts [A], Possible set of cases for a lesson in Telugu language primer [B]
for the scale and variety of 22 Indian languages and varied instructional designs. This specific knowledge has been abstracted into instructional design knowledge and patterns as shown in Chapter §3.

Generalizing from this specific knowledge, the domain knowledge representation in this thesis should be:

• in synergy with instructional design

• machine-processable

• facilitate reuse and semi-automatic design of applications (eLearning Systems)

• able to support sharing of knowledge between different applications and tools

More specifically, the focus of this thesis is to represent a subset of domain knowledge towards developing families of eLearning Systems for adult literacy in India rather than on discovering methods for representing entire instructional design or language learning.

Ontologies is one of the widely used methods to represent domain knowledge that suits the above requirements. Over the years, ontologies have garnered mainstream attention for its wide range of applications in several domains [143] [204]. However, the term ontology itself has several connotations and is understood in a variety of ways in the literature [143][205][206]. A commonly used definition of ontology in computer science comes from Gruber [9], where he defines an ontology as “a formal, explicit specification of a shared conceptualization”.

A common characterization of an ontology by several researchers [143] [205] as:

• formal, meaning that an ontology should be represented using a formal language processable by machines and tools

  This characteristic will allow us to represent instructional design knowledge in machine-processable form to facilitate automation

• explicit, by using different types of primitives and precisely stating different concepts and axioms defining the ontology

  Instructional design for adult literacy is embedded in IPCL and primers; ontologies will help in making it explicit

• shared, the ontology is meant for a group of stakeholders within a community belonging to a specific domain or sub-domain

  How to share IPCL across all Indian languages and How to specify this knowledge to software engineers?

• conceptualization, represents a specific view of a domain through various abstractions a view of instructional design for adult literacy in India
Fensel attributes the popularity of ontologies is due to the promise of providing “a shared and common understanding of a domain that can be communicated between people and application systems” [207], which can be construed as communication and automation requirements needs of this thesis.

4.2 A spectrum of ontologies

There has been a significant growth of research in ontologies in the past decade resulting in a variety of approaches for ontological engineering [208][209][68][146][210][147]. In particular, ontologies have been used for representing different aspects of a specific domain using a wide range of different mechanisms [211]. At the core of ontologies is the identification of concepts generally represented as a hierarchy of classes and sub-classes and different relationships between them. Each of the these classes typically have associated properties and can also have a set of constraints. Instances of these ontologies are called as individuals and represent a specific instance of a particular domain represented by the ontology. For example, an ontology for instructional design at a higher level can be defined using concepts like goals, process, content and so on and each of these sub-classes can be further defined. An instance of this ontology can be a specific instructional design for teaching a specific course. We discuss this ontology in detail in the rest of the chapter.

Diversified needs emerging from different domains gave raise to a spectrum of ontology kinds [211][148] as shown in Figure §4.3. These kinds of ontologies vary based on the degree of specification detail, formalism and expressiveness power as we move from one end to the other end of the spectrum. A detailed description of this spectrum is given in [211][206]. In essence, there are informal or lightweight ontologies on one end, primarily
geared towards some sort of communication and on the other end, formal ontologies help in automated reasoning of knowledge [148]. This thesis falls in the middle and mostly uses OWL/XML Schemas to address the primary needs of communication and automation. They also provide a mechanism to use instructional design as a basis throughout the design of educational technologies. In addition, two key future research directions that motivated the need for ontologies in this thesis are:

- Design of personalized learning environments for a diversified range of learners, teachers and subjects in India and across the globe
- Design of technologies that allow students to explicitly justify their answers through reasoning and provide a debugging environment through automated reasoning

Additionally, scope is another critical factor that can be used to classify ontologies at different levels of granularity. A distinction is made between upper ontologies that describe general-purpose concepts and their relationships, domain ontologies that define domain-specific concepts, task ontologies that specify domain-specific activities and application ontologies that instantiate domain ontologies and integrate task ontologies for a particular application [212].

We use OWL 2, a W3C recommendation that refines and extends OWL, the Web Ontology Language for representing knowledge in the semantic web [213]. OWL 2 is based on earlier version of OWL, extends RDF and is also compatible with XML. According to Krotzsch, OWL serves as a descriptive language for expressing expert knowledge in a formal way and as a logical language for drawing conclusions from that knowledge [214]. Accordingly, OWL 2 allows ontology engineers to represent knowledge using various representations like RDF/XML, OWL 2 XML, Functional Syntax, Manchester Syntax, Turtle as shown in Table §4.1 with each of the methods having different expressive power and reasoning abilities [213]. The choice of ontology representation is primarily decided by ontology engineers based on the requirements and needs of the domain [212]. We confine ourselves to the descriptive use of ontologies and use OWL/XML for representing ontologies in this thesis.

### 4.3 Development of Ontologies

Ontology development has matured in the last few decades from being a research topic to even the discipline of ontology engineering [209] and is often seen as one of the cornerstones of semantic web today [213]. There are several surveys in the literature focusing on ontological development methodologies [215] [216] [217][218]. Most of these methodologies like Cyc, METHONTOLOGY, TOVE are based on standard development processes described in terms of phases with several activities in each of the phases. A deeper analysis
Table 4.1 Syntax variations in OWL 2 Web Ontology Language

<table>
<thead>
<tr>
<th>Name of Syntax</th>
<th>Specification</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDF/XML</td>
<td>Mapping to RDF Graphs</td>
<td>Interchange (can be written and read by all conformant OWL 2 software)</td>
</tr>
<tr>
<td>OWL/XML</td>
<td>XML Serialization</td>
<td>Easier to process using XML tools</td>
</tr>
<tr>
<td>Functional Syntax</td>
<td>Structural Specification</td>
<td>Easier to see the formal structure of ontologies</td>
</tr>
<tr>
<td>Manchester Syntax</td>
<td>XML Serialization</td>
<td>Easier to read/write DL Ontologies</td>
</tr>
<tr>
<td>Turtle</td>
<td>Mapping to RDF Graphs, Turtle</td>
<td>Easier to read/write RDF triples</td>
</tr>
</tbody>
</table>

of methodologies for building ontologies is presented in [215] and from an evaluation perspective in [219]. There is a consensus in the literature that none of the existing approaches are fully mature when compared with software engineering and knowledge engineering methodologies [216] [220]. METHONOLOGY is one of the oldest and matured approach for ontology development [215] but without support for collaboration. A software engineering approach for developing ontologies is proposed in [221]. A six-stage ontology development method for engineering design was proposed by Ahmed et al. in [222]. Ontology development methods supported with tools became quite popular in the recent times and protégé is one of the exemplar examples to illustrate this case. In their Ontology Development 101, Noy and McGuinness proposed an iterative approach for building ontologies consisting of several activities that need not be sequential (i) determine scope (ii) consider reuse (iii) enumerate terms (iv) define classes (v) define properties (v) define
constraints (vi) create instances. An important conclusion from their work is “there is no single correct ontology for any domain. Ontology design is a creative process and no two ontologies designed by different people would be the same” [223].

We see three major directions for developing ontologies from a synthesis of the literature (i) manually by expert(s) for specific purposes following a varied set of processes from lightweight to a rigorous standard process (ii) semi-automatic way of developing ontologies, where a part of the ontology is developed manually and a part is automatically retrieved using text mining, natural language processing and other machine learning techniques (iii) fully automatic development, where the ontologies are derived using ontology learning approaches.

We follow a simple process for developing ontologies in this thesis as shown in Figure §4.4. The first step in the process is to determine the requirements from the ontology, which is driven by the set of eLearning Systems to be developed in our case. The next step is to figure out the scope of the ontology drawing a boundary for what is within and outside the scope. Once the scope is defined, the next step is to identify any existing ontologies that can be used for creating the ontology. There are several search engines like SWOOGLE\(^1\) and ontology servers like OntoLingua\(^2\)’s for searching existing ontologies. We discuss the ontologies we adapted from existing literature in the next section. Once the suitable sources for ontologies are defined, the next step is to use a standard approach to identify the concepts, relationships between the concepts, define properties, constraints and instances using an appropriate representation language like OWL/RDF. An important distinction of this process from the standard ontology development methodologies is the use of patterns as one of the critical sources for building ontologies. The patterns themselves are discovered after extensive discussions with domain experts; rigorous analysis of literature and analyzing existing applications that are built in the domain. We have extensively discussed with domain experts from NLMA; analyzed literature on adult literacy and instructional design as a source for our patterns. We also analyzed several eLearning Systems developed by TCS for 9 Indian languages before creating the patterns. We use these patterns as one of the primary source for creating the ontologies. We also consider other literature from the instructional design space as input to our ontologies. The output of this entire exercise of conceptualization and implementation is a set of ontologies. The evaluation of this ontologies is carried out by developing a set of applications based on these ontologies and assessing whether the domain requirements have been met or not. Figure §4.5 shows a part of how we devised scope for our instructional design ontology framework for adult literacy.

\(^1\)http://swoogle.umbc.edu/
\(^2\)http://www-ksl.stanford.edu/knowledge-sharing/ontolingua/
4.4 Ontologies for Instructional Design

Ontologies are used in education domain for several applications ranging from representation of subject knowledge [224] to generation of content and assessment artifacts based on learners' styles [50] [151] [225]. Researchers have extensively worked on building systems based on ontologies primarily in the area of adaptive and personalized learning [150][226][227]. Ontologies are also applied in intelligent tutoring systems [228][229] [230] and for automatically generating multiple choice questions from domain ontologies [231]. Mizoguchi suggested several ways of how ontologies can help in addressing challenges in AI-ED domain [151] and reflected on their progress after 15 years in [210]. He suggested that there is long way to go for advancing methods for ontology development and their use in education [210]. Synthesizing the literature in this space, we see different kinds of educational knowledge [36] that can be modeling using ontologies:

- Curriculum, selection of knowledge
- Courses, selection of specific knowledge pertaining to a subject
- Instructional Design, how this knowledge is taught
  - context
  - goals
• Evaluation or Assessment, to figure out if knowledge is acquired as per goals

• Learner Styles, knowledge of different kinds of learners

• Teacher Styles, knowledge of different kinds of teachers

We do not attempt to model this entire knowledge but a subset of this knowledge within the scope of the thesis. We are interested in the development of ontologies for instructional design in the context of adult literacy in India. The key purpose is to create variants of instructional design catering to the varying needs of eLearning Systems for adult literacy. We presented different categories of patterns in instructional design in Section §3.5 and in this section we present ontologies based on these patterns for different aspects of instructional design. We have extensively searched the literature to find ontologies that are relevant for our purposes.

What are the existing ontologies related to adult literacy in India?, and the domain of instructional design in general?

To the best of our knowledge, we could not find any ontologies that are even remotely connected to adult literacy in India. But we searched the literature for various ontologies focusing on different kinds of educational knowledge and give a few examples here. An ontology for literacy was proposed in the context of intelligent tutoring systems way back in 1999 [232]. We then looked into some upper ontologies and found an example curriculum ontology devised by BBC for the national curricula on UK focusing on three topics (Algebra, Geometry, Formula), level (KS1, KS2, KS3, GCSE) and different fields of study (Maths, English, Science) [233]. A comprehensive ontology that models several learning theories is presented in [234] where the idea is to have solid pedagogical basis for intelligent tutoring systems. Recently, Heiyanthuduwe et al. have analyzed 14 ontologies developed by different institutions for learning design and proposed an OWL 2 learners profile [235]. One of the earliest ontologies developed by Mizoguchi focuses on creating a task ontology to facilitate reuse of problem solving knowledge [67]. We came across several ontologies focusing on particular kind of instructional design; for example, a mobile learning ontology was designed for abductive science inquiry style of instruction [236]. An ontology for learning scenarios based on collaborative learning theories is given in [237] and one focusing on gamification is presented in [238]. There were other set of ontologies focusing on specific subject matter, like word problems in mathematics [239], software engineering body of knowledge [240]. In addition to these kinds of ontologies, there are
different kinds of ontologies developed for learning content [156], learning design based on IMS LD standard [42], a context ontology for bridging the gap between learning content and learning design [241]. There were ontologies to represent learning object repositories [242] and learning design repositories [243] to facilitate search and retrieval of learning resources on the web.

However, none of these ontologies are connected to IPCL or patterns and hence do not directly meet the requirements in this thesis. We will discuss an ontology based modeling framework in the next section followed by domain ontologies for adult literacy.

### 4.5 IDont - An Ontology Based Framework for Modeling Instructional Design

Is teaching science the same as teaching mathematics? Does teaching in a country like US and India same? Can we use the same method for teaching different kinds of learners? The answer for most of these questions is no. There has been tremendous effort in trying to come up with several standards in the space of educational technologies like SCORM for learning objects [244], IMS-LD for learning designs [181], IEEE LOM for learning objects [46] and enormous research [183], platforms and tools [32] surrounding these standards. However, despite significant progress, most of the promises seem to be unfulfilled as discussed in Section §2.4.2 and in Section §2.4.3. We summarize the following major pitfalls:

- **One-size-fits-all** - There are hundreds of learning theories in the literature. Attempts towards coming up with a unified way of dealing with them turned quite complex denting the success of the initiatives.

- **End-to-end automation** - Several attempts have been made to automate different aspects of education, ranging from modeling learning theories to automatically generating learning environments and this focus on end-to-end automation turned futile in the most of the attempts [182]. For instance, in the case of IMS-LD, even though not stated explicitly, this goal of end-of-end automation resulted in complex authoring [183].

- **Administration and Management** - While it is important to handle and ease the job of teachers in administration and management activities for which several Learning Management Systems were developed, linking the instructional design to LMS has increased further complexity with the existing approaches.

We learn from these experiences and propose a framework for modeling instructional design using ontologies based on patterns. The design rationale for IDont is as follows:
• **Simplicity & Separation of concerns approach** - We strongly advocate the separation of concerns approach to model ontologies. The core idea is to have smaller multiple ontologies for different aspects of instructional design such that they can be adapted, extended and reused in other contexts. Figure §4.6 shows how different aspects can be separated as components having explicit interfaces such that they can be connected with other aspects and customized for a specific learning situation.

• **Leverage and Reuse existing ontologies** - When designing new ontologies, ontological engineering suggests the utilization, adaptation and extension of existing ontologies (learning objects, learning designs).

• **Technology Design** - The framework should support the creation of a platform and authoring tools to explicitly capture and model all the ontologies.

• **Extensibility and Customization** - These are two major criteria for the framework as most of the times the ontologies have to be customized and extended for the specific domain. There should be a provision in the framework such that existing ontologies should be replaced with custom ontologies with minimal effect on the overall framework.

• **Iterative and Collaborative Approach** - The design of this framework should follow an iterative approach and must consult different stakeholders (such as teachers, learners, instructional designers and so on) during the process.

• **Internationalization** is required for both ontologies and tools that support the creation of ontology instances.

The core premise of this framework is to systematically model instructional design using different aspects like context, goals, process, content, evaluation, environment. We distinguish between two kinds of instructional design knowledge, one is at a conceptual level that maps with existing learning methodologies and the other is at a technical level to facilitate semi-automation of eLearning Systems. In this context, the core idea of IDont is not to define complete ontologies but to point to several possible modular ontologies that are required for systematic modeling of instructional design. As such, most of the aspects of IDont are optional and can be configured based on specific purposes and learning situations. Figure §4.6 shows an overview of the IDont framework. The key inputs come from a set of instructional design requirements that drive selection of appropriate instructional design models, which are captured as patterns in our approach. We do not specify the exact ontologies for instructional design but have placeholders for different aspects. With the advent of several ontology repositories, an instructional designer or ontology engineer can either extract the required ontologies for specific instructional design model from existing literature or create a new one. This generic instructional design stitched from existing or new ontologies can
Figure 4.6 Overall Process of IDont Framework
be customized with domain ontologies and is further realized by specific instances like ID1, ID2 ... IDn.

Figure §4.7 presents an overview of IDont framework for adult literacy. Even though we discuss several ontologies in the diagram, the focus of this thesis is on goals, process and content ontologies. We briefly explain the important ontologies of our framework as follows:

A. **ContextOntology** - Context plays a significant role in IDont as it allows for modeling of various aspects related to a particular learning situation. The notion of learning context was proposed in LOCO ontology to bridge the gap between learning design and content consisting of domain specific information [43]. However, in this framework, we articulate context in a broader view that encompasses several pointers to all other ontologies. This is a meta-ontology that essentially captures the basic information related to all other aspects of instructional design such that each of these aspects can be potentially (re)used. As shown in Figure §4.7, ContextOntology has metadata associated with it along with context information related to various aspects of instructional design. ContextOntology specifies how a ProcessContext achieves goals using ContentContext delivered through EnvironmentContext following EvaluationContext and performed by RolesContext.

B. **GoalsOntology** - This ontology formalizes the notion of goals (which can be instructional goals, learning goals or even learning outcomes). The details of how it is defined are left to the specific instance. Some properties associated with goals are hasName, hasPriority, hasPrerequisites, hasEvaluation, isAchievedByProcess. The GoalsOntology points to the process through which these goals will be achieved, target competencies, the instructional material that is required and the evaluation to be performed. Consider the scenario of creating goals for K12 students, and goals are prescribed by education boards. Teachers can potentially reuse these goals if captured in the form of GoalsOntology. As the evaluation related to these goals is separately captured, it can be reused as well. We prescribe to the idea of goal-driven instruction as part of our framework irrespective of instructional design models. We detail GoalsOntology in the latter part of this chapter in Section §4.5.1.

C. **RolesOntology** - The success of instructional design depends on several people who perform their roles in the process. Hence, it is important to capture the roles, their responsibilities and how the different aspects of instructional design should be adapted according to the needs of specific roles. The two most important roles are that of teacher and learner. The crucial knowledge about a learner is captured through learner profiles consisting of several attributes. Generic roles in instructional design are captured first and then modified based on specific learning situations. Mapping of goals with competencies of roles can also be done in this ontology. Most of the roles are further associated with teams and in that case the role of the team as well as individual persons is modeled separately along with roles performed by agents. This ontology should also have metrics to trace from goals to evaluation.
**D. Process Ontology** - The crux of IDont framework is the Process Ontology that captures the instructional design process, and relates to all other ontologies and practically executes the process. In the literature, Learning Design is discussed heavily, in particular IMS Learning Design [37] and received criticism as well [183]. Ontologies for modeling learning design are presented in [42]. Based on our prior experience with adult literacy instructional design, IPCL and our future goal to introduce reasoning into adult literacy, we proposed the Process Pattern - (play, act, scene and instruction) in Section §3.5.2. Each lesson is organized as hierarchy of pasi with instructions where concrete activities are performed based on Merrill's principles of instruction in this particular instance. This instruction actually points to Content Ontology and associates required content for the respective instruction. This nomenclature allows us to systematically capture the knowledge of instructional design process and potentially reduce technological effort. This hierarchy has similarities to IMS LD but has variations to align with our patterns for adult literacy instruction. We will present the Process Ontology in Section §4.5.2.

**E. Content Ontology** - This ontology allows for modeling of instructional material in a particular learning situation. There is extensive research on ontologies for learning objects and we use the ALOCoM ontology [69] as base for content aspect of our framework. However, for adult literacy instructional design, we have used fcrmt (facts, cases, rules, models, and theories) structure [7] as discussed in Section §3.5.3. So the ContentType of ALOCoM now includes fcrmt. The Content Ontology is closely associated with other ontologies and strongly with the Process Ontology.

**F. Evaluation Ontology** - What if the most common evaluations of instructional design are captured and an instructor can customize them based on his or her requirements? The main intent of this ontology is to capture evaluations as independent knowledge and link them with goals through Context Ontology. This separation makes it easier to perform different kinds of evaluations for the same set of goals. This ontology captures the details of evaluation and has a direct relationship with Goals Ontology which is connected with Process Ontology.

**G. Environment Ontology** - The final execution of instructional design happens in an environment. Consider the scenario of a lesson to teach about shapes to K2 students. Considering that the students first listen about shapes in a Classroom Environment and if a teacher wants to use computers, then most of the other aspects of instructional design remain same except the environment which changes to Computer Environment. Separating the environment from the rest of instructional design makes it easier to run the learning situation in different environments similar in spirit with software deployment.

**H. Domain Ontology** - This ontology mainly articulates and customizes key aspects of instructional design with respect to a specific domain and provides a domain-specific version of ontology. In particular, the various sub-ontologies and properties of these ontologies will have detailed associations when mapped to a specific domain. For e.g. the Content Ontology will have strong co-relation and mapping with content in the domain.
There can be several other ontologies like ActivitiesOntology for capturing various activities that are supported in the instructional design, WorkflowOntology to model the tedious workflows in education, FeedbackOntology to capture continuous feedback of the instructional design, OrganizationOntology focusing on characteristics of the organization, ResourcesOntology, having pointers to specific resources like text, audio, video and so on. In our analysis of instructional design literature, we strongly see that it is virtually impossible to capture all kinds of instructional design models and any attempt towards it turns to be futile. However, the main intent of our framework is to use a separation of concerns approach to systematically capture various aspects of instructional design through ontologies.

*The primary goal of ontologies in this thesis is to drive the semi-automatic creation of eLearning Systems based on accepted instructional design models.*

We discuss the specific ontologies that are developed as part of IDont framework. Our attempt is never to be complete during the design of these ontologies but to design educational technologies in sync with instructional designs. We also include several entities in the
ontologies for future use rather than just current needs. We also rely on the best practice of adapting existing ontologies and create new entities only if required.

### 4.5.1 An ontology for modeling goals

The primary goal of any instructional design is to find ways to support learners in achieving their learning goals [245]. Based on the pattern discussed for goals in Section §3.5.1, we present an ontology for representing instructional goals in this section based on revised Bloom’s taxonomy [14].

Figure §4.8 shows a part of GoalsOntology developed using protégé\(^3\) tool from Stanford. The priority of the goal is described using GoalPriority, progress through GoalProgress, deadline through the property goalDeadline. An important sub-class is to classify the goal according to a taxonomy. The class GoalClassification is further divided into two classes BloomTaxonomy and ABCD. The BloomTaxonomy is further divided into KnowledgeDimension and CognitiveProcessDimension as per revised Bloom’s taxonomy. The knowledge can be classified as FacetualKnowledge, ConceptualKnowledge, ProceduralKnowledge and MetaCognitiveLevelKnowledge with increasing levels of higher order levels of thinking. This is in sync with the ContentPattern discussed in Section §3.5.3. The CognitiveProcessDimension is the most commonly used way to classify goals as per Bloom’s taxonomy. It has six levels Remember, Understand, Apply, Analyze, Evaluate, Create and each of them have sev-

\(^3\)http://protege.stanford.edu/
eral verbs specifying the activities. Several object properties are shown in Figure §4.8 connecting different concepts in the ontology. Priority of the goal can be captured using goalPriorityLevel, competency through goalCompetencyLevel and goalKnowledgeLevel can have a range of values from the KnowledgeDimension and maps to the fmt pattern. Every goal should have a goalDeadline and its progress is monitored through goalProgress. A goal also has hasPrerequisites, previousGoal and nextGoal. This ontology is connected to ProcessOntology through isAchievedByProcess, ContentOntology via usesContent, EvaluationOntology through hasEvaluation and runsInEnvironment. In addition, there are several data properties that are associated with the ontology. For example, goalDeadline stores the deadline as dateTime. The goal itself can be described using goalText, goalImage, goalAudio, goalVideo, goalMetadata. These data properties store specific information that can be later used for (semi-)automatic generation. GoalGranularity is another critical class that is specific to our instructional design as we have a goals hierarchy akin to play, act, scene, instruction pattern. In addition to the standard concepts, the ontology also has concepts for GoalPattern consisting of properties shown in Figure §4.8. For example, SourceOfPattern is a data property that specifies the source of the patterns, Trade-Offs specifies the issues that might occur using this pattern. In our case, we realized that if specifying goals requires so much of effort, the entire exercise will be a burden for teachers and instructional designers making it a futile effort in the end. Hence, we have minimal mandatory properties with scope for using extended properties only if required. A detailed listing of this ontology is provided in Appendix §A.1. We will provide instances of this ontology in Chapter §6 as part of evaluation.

4.5.2 An ontology for modeling instructional processes

The ProcessOntology is a core ontology for specifying instructional process and is closely associated with several other ontologies. As shown in Figure §4.9, the ontology is divided into three conceptual sections at a higher level (i) learning, focusing on concepts that map to the underlying learning methodologies (ii) metadata consists of information about the process in general (iii) user interface to declaratively specify a few aspects of the eLearning System. The ProcessOntology is based on ProcessPattern and its primary purpose is to achieve goals specified in the GoalsOntology and is connected through hasAssociatedGoal property. These goals have to be achieved using content specified via ContentOntology connected through the object property usesContent. Similarly, usesEvaluation, performed-byRole, runsInEnvironment connect this ontology to EvaluationOntology, RolesOntology and EnvironmentOntology respectively. This ontology has several data properties like title, description, metadata, noOfPlays, noOfScenes, noOfInstructions. One important property is hasTimeLimit that specifies the time limit for an activity, instruction, scene, act, play. Guidelines is an important concept that we use for giving instructions to learners during their interac-
tion with the eLearning System at different levels of granularity specified using PlayGuidelines, ActGuidelines, SceneGuidelines, InstructionGuidelines, ActivityGuidelines. For example, a guideline from a teacher might be “Everybody look at the screen and observe how the two syllables are combined together to form a new word”. Separating this information provides the flexibility to change guidelines. This can be specifically used to change medium of instruction in an eLearning System. A language like Hindi can be taught using Telugu as medium of instruction by changing the guidelines in the entire system.

The base InstructionalDesignModel can be specified as MerrillModel or any other instructional design model from the literature. We use MerrillModel for the reasons motivated in Section §3.5.2. Then each lesson is modeled using a set of plays (GenericPlay) that are divided into acts (GenericAct), which are further divided into scenes (GenericScene) and instructions (GenericInstruction). We have identified different kinds of acts for adult literacy instruction as specified in the ProcessPattern in Section §3.5.2. They include a MotivatingAct, NewPhonemesAct, FormingWordsAndSoundsAct, SyllableBankAct, ComparingAct, LearningRulesAct, WritingInstructionsAct, ExerciseAct, SummaryAct. We inferred these acts from IPCL and eLearning Systems that are tested on the field. There are different kinds of scenes SimilarSoundsScene, SimilarSyllablesScene, InspectingSyllableBankScene, SyllableFormationRulesScene, FamiliarWordsScene, SyllableBannerScene, FormingWordsScene under each act. Each scene further has instructions which have direct activities for facilitating learning. Each instruction follows one or more principles and can have one or more activities. We specify Merrill’s first principles of instruction using FirstPrinciples that is fur-
ther divided into IntegrationPrinciple, ActivationPrinciple, DemonstrationPrinciple, ApplicationPrinciple, DemonstrationPrinciple. Activity is one of the most commonly used concept in the space of instructional design and we model that using GenericActivity. We incorporate two kinds of activities from the literature LearningActivity and SupportActivity. But we also model four kinds of additional activities StructureActivity, GuidanceActivity, CoachingActivity and ReflectionActivity to accommodate Merrill’s inner circle of structure-guidance-coaching-reflection. Modeling these activities as concepts allows us to change these activities based on learner styles or instruction styles. In addition, InterpretedActivity and MoniteredActivity help from evaluation perspective.

The current ontology also has basic concepts for UserInterface like AnimationStyle, ColorTheme, AnimationSpeed, Language, Background. The instances of these concepts will help in configuring the user interface of eLearning Systems for adult literacy based on specific requirements.

One principle behind this ontology is not to use all the classes and properties but to further filter this ontology to the specific needs and use only a fragment of the ontology for the purpose in order to reduce the burden on the teachers and instructional designers. For example, if a course has 1000 instructions in total, then specifying principles for all of these instructions might be a burden and an alternative could be to make this property optional at instructional level but mandatory at a scene or act or play level. A detailed listing of this ontology is provided in Appendix §A.2.
4.5.3 An ontology for modeling instructional material

This ontology is primarily derived from existing literature on learning objects and specifically the ALOCoM ontology [156] along with the ContentPattern discussed in Section §3.5.3. As shown in Figure §4.10, this ontology includes four core concepts Content Type, Content Fragment, Content Object, Learning Object. The raw data in the form of text, audio, animation, video are concepts in Content Fragment and Content Object is an aggregation of several content fragments. The way our ontology differs from literature is in terms of Content Type, which includes Facts, Cases, Rules, Models, Theories, which form the Core Type. In Extended Type, there are further concepts derived from the literature [156]. Essentially, they capture learning objects at a higher level of abstraction. Another important concept is Learning Object which has the sub concepts of Play Object, Act Object, Scene Object, Instruction Object. These concepts are connected to respective elements in Process Ontology.

There are other ontologies for specifying Roles, Evaluation, Environment that are part of instructional design ontology but defining those ontologies is beyond the scope of this thesis. The Roles Ontology is an interesting one with roles like Teacher, Mentor, Teaching Assistant, Coach and so on and can be mapped to different kinds of activities in the Process Ontology. As an example, learning styles and teaching styles may be used in the role of learner and teacher in Roles Ontology. A detailed listing of this ontology is provided in Appendix §A.3.

In the next section, we will briefly discuss a domain ontology for adult literacy.

4.6 A Domain Ontology for Adult Literacy

A distinction has been made in the literature between domain and application ontologies [212]. Domain ontologies are aimed towards defining concepts pertaining to a specific domain like “adult literacy for Indian languages” whereas application ontologies are further refined to specific needs of an application, in our case it can be an eLearning System for a particular language. According to Census 2011, there are about 29 languages spoken by more than a million people, 60 languages by more than 100,000 people and about 122 languages by more than 10,000 people speaking it in India. Of these, there are 22 official languages. A fundamental tenet of Indian languages is that they share a common phonetic base [246]. This commonality across a family of Indian languages provides a shared domain that can be used for representing domain knowledge as an ontology. Based on this premise, NLMA has come up with IPCL, a uniform methodology as explained in Section §2.1, which is the base for creating instructional material for all Indian languages. In Indian languages, the term “aksharas” is used to refer to alphabet. This akshara refers to a sound that is formed of sounds of vowels and consonants [247]. Being invariant of position is an interesting characteristic of akshara that holds for all Indian languages [247].
Figure 4.11 A fragment of Domain Ontology for adult literacy

Figure §4.11 [left] shows an ontology of Indic scripts for literacy, primarily focusing on the structure of syllables in the language. The core concepts of the ontology include Syllable denoting the visual representation of an akshara or a fragment of it. This is further specialized into SimpleSyllable, CompositeSyllable and SpecialSyllable. Every concept in this ontology have two properties hasRepresentation and hasResources refering to Representation and Resources respectively. Representation has five other concepts to systematically capture a syllable. Figure §4.11[right] shows the core structure of a syllable in Indian languages. The syllable itself is composed of CoreSymbol, LeftSymbol, RightSymbol, TopSymbol, BottomSymbol. Each of these concepts store the respective visual fragments of the syllable at the relative positions. For example, a base consonant like क can be modified using any of the vowel modifiers from left ि to give कि, adding the right symbol िी to क results in की. Similarly, adding top symbol ए gives के and bottom symbol उ results in कु. The composition of these symbols is represented as CompositeSymbol. The most important property of Indian languages is the broad choice of each of these symbols giving rise to an entire alphabet for a particular language. Figure §4.11[right] shows how vowel modifiers can be applied on four sides of a base consonant to give a set of composite symbols in a language. For most of the Indian languages, the number of vowel modifiers is 12 eventhough there are a few languages where the number can be less. Similarly, the number of consonants will vary from 12 to 36 for different languages.

The concept of Resources in the ontology helps in storing the data for respective syllables and phonemes in the form of Text, Audio, Image. We have observed that most of
the eLearning Systems for Indian languages developed today rely on images for storing and displaying language aspects making it difficult to change the system. However, in our ontology we make an attempt to systematically separate different aspects to facilitate variety for a multitude of languages. The ontology also has the concept of Vowel, that is further divided into SimpleVowel and CompositeVowel. They are bound to Syllable through the property hasVowel. Similarly, Consonant class represents the consonants in a given language and can be either SimpleConsonant or CompositeConsonant. Most of the Indian languages have ConsonantConjuncts like ḷ, ḷ, ʈ, ṭ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ in Telugu language, which are special symbols formed with a number of syllables. Modifier is the core class for representing different modifiers like VowelModifier and SpecialModifier. VowelModifier in general consists of several signs often called as dependent vowels ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ, ḷ. In Devanagari, there are twelve signs which when composed with consonants give rise to a number of composite syllables and is often called Barakhadi because of 12 modifiers. In addition to these, there are several special modifiers specific to a language. We have modeled Phonemes in similar lines to Syllables through respective concepts. Numerical class denotes the representation of symbols ० १ २ ३ ४ ५ ६ ७ ८ ९ in a particular language. We did not get into writing part in detail even though we have left scope for further extension of the ontology. This entire exercise of creating detailed ontologies for adult literacy is to systematically specify different parts such that they become source of variety to facilitate semi-automatic development of eLearning Systems. A detailed listing of this ontology is provided in Appendix §A.4. In the next chapter, we will discuss how we have harnessed the common nature of families of Indian languages for design of eLearning Systems for adult literacy.