

CYN-D AND BURC

Simulation and Knowledge Acquisition using ResearchCyc

By Kino Coursey at Daxtron Labs
and The University of North Texas

(kino@daxtron.com)

OVERVIEW

- CyN-D: Simulation
 - History
 - Goals of Research
 - Architecture of CyN-D
 - Applying Cyc to AIML and Doom
- BURC: Acquiring Common Knowledge
 - What is Common Knowledge?
 - Mining with ResearchCyc
 - Results of Experiments
 - Next Steps
- How They Fit Together
- Benefits

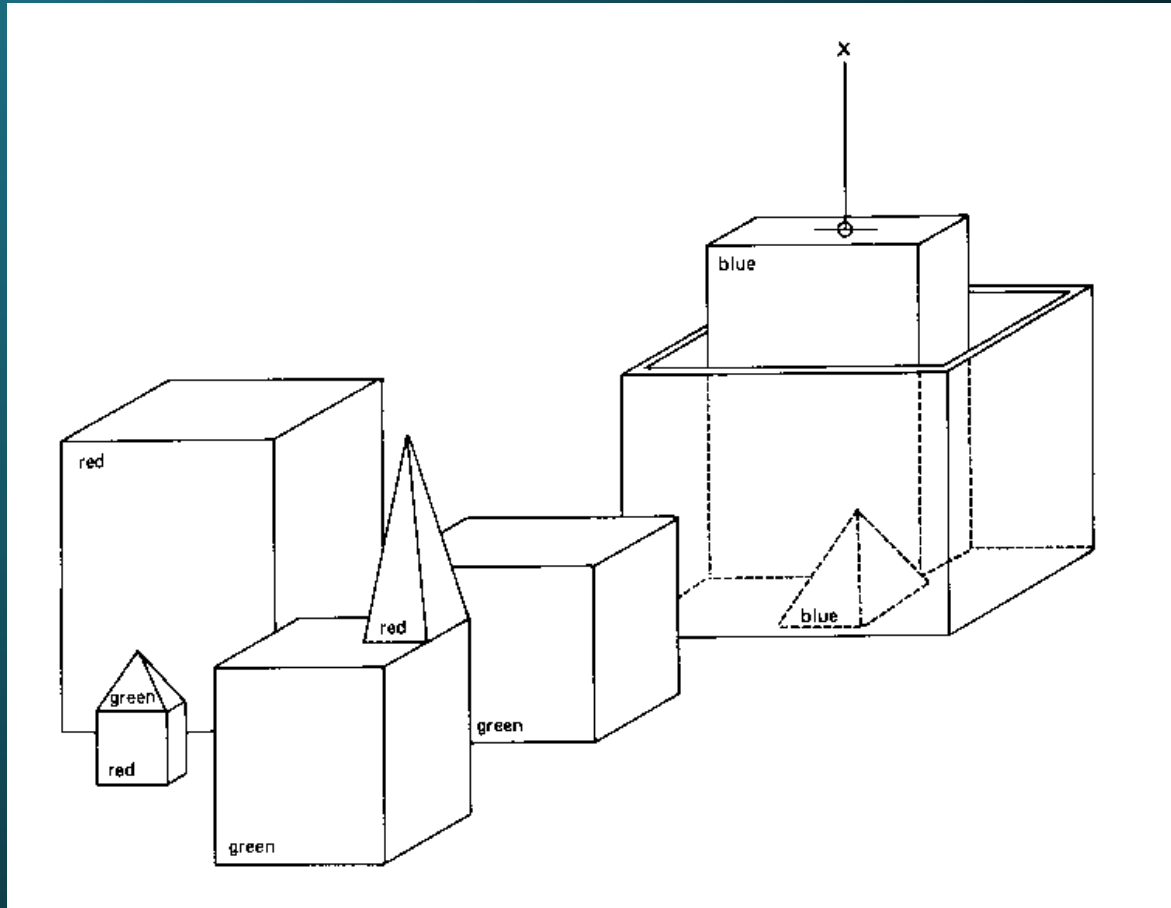
CYN-D AND BURC GOALS

- Develop applications using Cyc that can:
 - Modify and expand the KB using natural language
 - Provide multiple interaction methods
 - Demonstrate integrating and embedding Cyc
 - Control internal and external processes
 - Given a goal, develop plans to achieve them
 - Be rapidly adaptable to new users and situations
 - Be aware of its capabilities and limitations
 - Explain its reasoning (more completely through multiple interfaces)

CYN-D BASIC IDEA

- Combine modern inference engines, natural language processing and simulation
 - ResearchCyc version of Cyc commonsense reasoning engine
 - CyN AIML driven language front-end
 - Doom3 game engine for 3D environment
- Develop an easy to use, interactive agent system in a 3D environment that is extendable
- Integrating all of them 'properly' is the real challenge
- *What would SHRDLU 2005 look like?*

THEN: SHRDLU (1970)



SHRLOU DIALOG

What does the box contain?

THE BLUE PYRAMID AND THE BLUE
BLOCK.

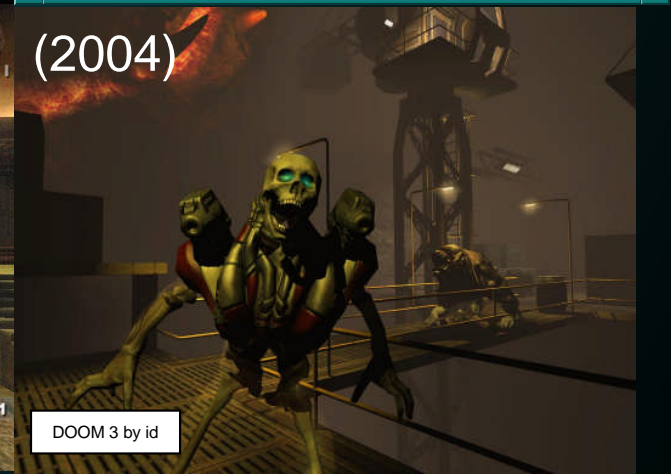
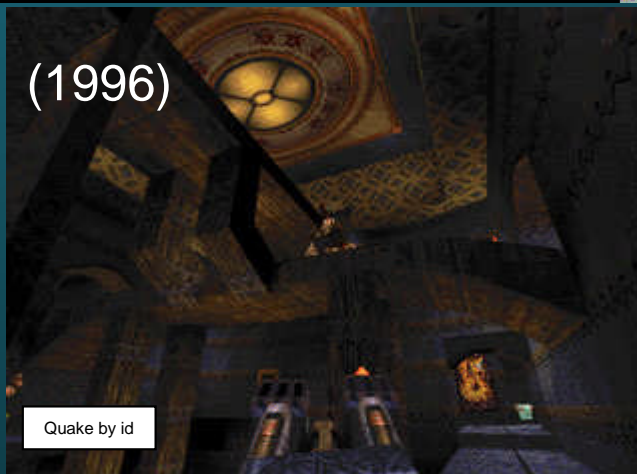
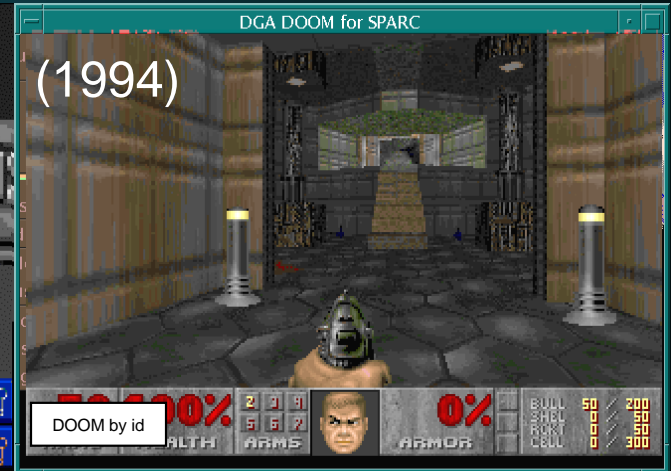
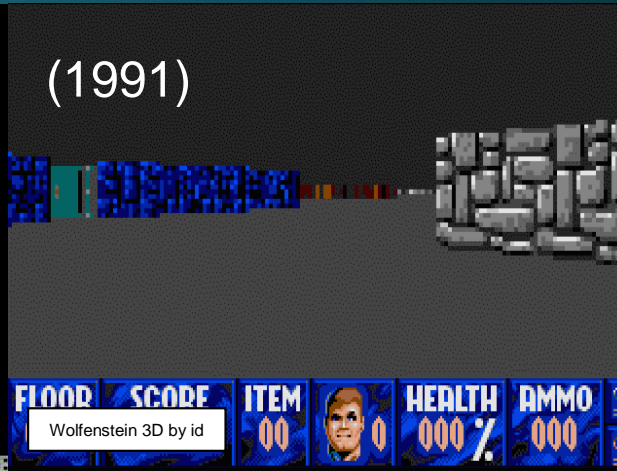
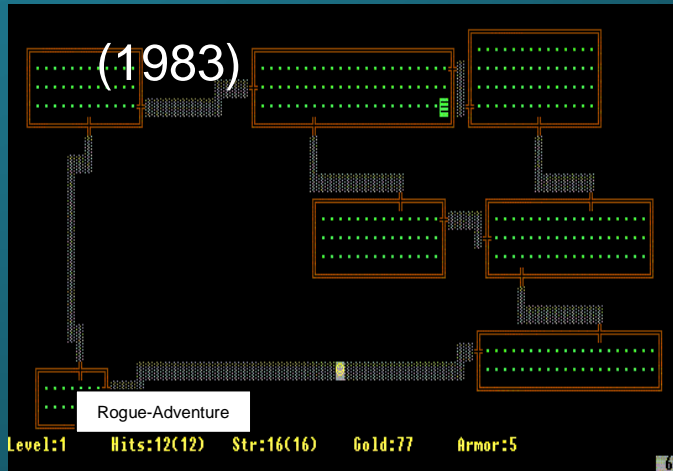
What is the pyramid supported by?

THE BOX.

How many blocks are not in the box?

FOUR OF THEM.

THE DRAMATIC IMPROVEMENT IN DESKTOP GAME REALISM



<http://agents.media.mit.edu/projects/videogame/>

NOW: CYN IN DOOM3 (2005)



CYN-D DEVELOPMENT GOALS - 1

- Provide a toolkit for experimentation with Cyc, AIML and Doom
 - Provide a pattern based parser for use with Cyc tools suitable for chatbots / virtual human experimentation and basic knowledge entry of logical relationships
 - Be an interpreter that allows an easy way to interface AIML to Cyc tools
 - Create a set of example AIML categories to answer questions using Cyc tools

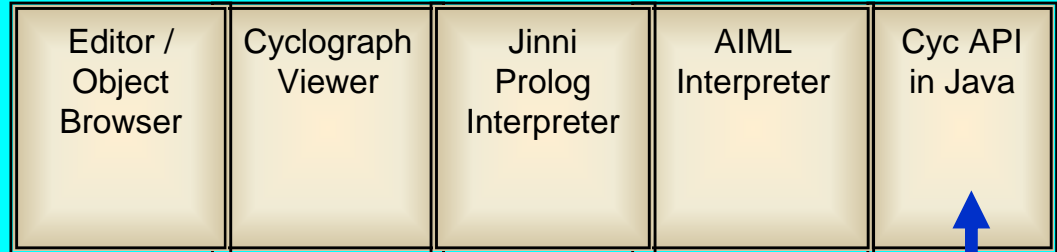
CYN-D DEVELOPMENT GOALS - 2

- Provide a natural language front-end ahead of the current Cyc tool release schedule
- Be a tool capable of creating a “personality” for Cyc systems
- Be scalable (AIML sets have 40,000 language patterns and full Cyc has 1.6 million facts)
- Be a tool that can be integrated with other systems
- Be low-cost and open
 - The original CyN is open source; Doom SDK is free
 - Cycorp adopted white paper “*Living in CyN : Mating AIML and Cyc Together in Program N*”
 - http://www.cyc.com/doc/white_papers/Cyn_description.pdf

CYN-D ENVIRONMENT

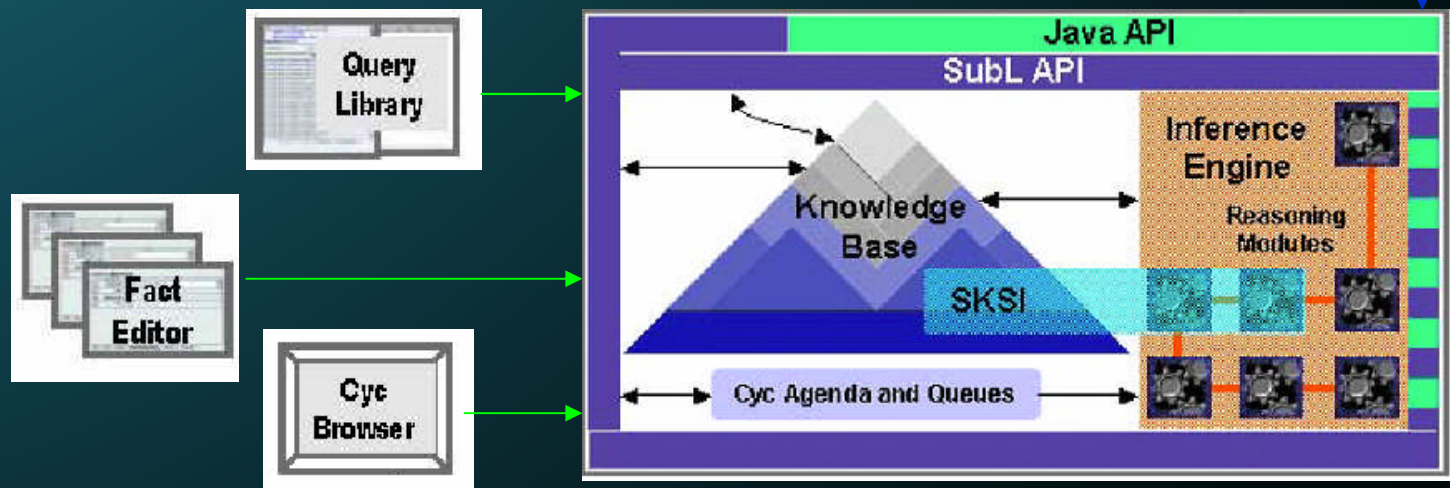
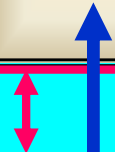


Doom3 Rendering Engine



Java Virtual Machine

Doom 3 World Simulation System



CYCLOGRAPH BROWSER

The screenshot displays the Cyclograph Browser interface, which is divided into several panels:

- Class browser:** Located on the left, it shows a hierarchy of classes. The "Object browser" section lists various classes such as `Class-InvocationHandler`, `Class-IClass`, `Class-IEntity`, `Class-IMover`, `Class-IActor`, `Class-IAI`, `Class-IPlayer`, `Class-IDoor`, `Class-ICamera`, `Class-Comparable`, `Class-IEntity-Proxy`, `IdEntity:player1`, `IdEntity:cyc_bot_2`, `IdEntity:cyc_bot_2_head`, `IdEntity:cyc_bot_1`, `IdEntity:cyc_bot_1_head`, `IdEntity:light_3`, `IdEntity:light_11`, `IdEntity:light_10`, `IdEntity:light_6`, `IdEntity:trigger_hurt_1`, `IdEntity:func_emitter_2`, and `IdEntity:func_static_3`.
- GLPanel-theGLPanel:** The central panel displays the details for the `SpatialPredicate` class. It shows the following information:
 - Collection:** `SpatialPredicate`
 - GAF Arg: 1**
 - Mt:** `UniversalVocabularyMt`
 - isa:** `AtemporalNecessarilyEssentialCollectionType`, `PredicateType`, `ConventionalClassificationType`
 - Mt:** `BaseKB`
 - isa:** `QAClarifyingCollectionType`
 - Mt:** `UniversalVocabularyMt`
 - gens:** `CotemporalPredicate`
 - Mt:** `BaseKB`
 - collectionConventionMt:** `UniversalVocabularyMt`
 - Mt:** `UniversalVocabularyMt`
 - comment:** "A specialization of `CotemporalPredicate`. Each instance of `SpatialPredicate` is a spatial relation that can (only) hold between one or more `SpatialThings`, and is used to state something about its/their spatial location(s), position(s), orientation(s), or dimensionality. Note that when a `SpatialPredicate` has a `Group` as one of its arguments, a certain
- Graph:** On the right, a network graph shows the relationships between various classes. A context menu is open over the `SpatialPredicate` node, listing actions such as "Expand Node", "Inspect Node", "Inspect GDS", "Inspect Google", "Inspect ANSWERBUS", "Inspect NEWS", "Inspect FROOGLE", "Inspect SCHOLAR", "Inspect IMAGES", "Collapse Node", "Hide Node", and "Center Node".
- Footer:** The bottom of the window contains a URL (`http://10.1.1.3:10702/cgi-bin/cyccgi/cg?cb-c-total&c11586`), navigation buttons (Zoom, Rotate, Radius, Hyperbolic), and dropdown menus for "GraphRoot" (set to "Agent-Generic") and "gens" (set to "isa").

WHAT IS AIML?

- “The HTML of Chatbots”
- Alicebot is the best known example, as the Lobner Prize winner. “Eliza 2005”
- Simple XML format for specifying surface linguistic patterns and responses
- Very easy to learn, hence its popularity
- Includes recursive decomposition and topic / context sensitive features
- Includes ability to compose and merge response sets from different sources
- Tags are translated into composable nano-agents
- CyN restores original capability to execute arbitrary programs in addition to generating responses

AIML USING CYC FOR Q&A

- AIML Code fragment activated by question

```
<category>
```

```
  <pattern>DO YOU LIKE *</pattern>
```

```
    <guard>
```

```
      (fi-ask '(#$genls <cycterm><star index="1"/></cycterm>  
              #$DangerousThing) #$EverythingPSC) </guard>
```

```
  <template>
```

```
    I don't play with <star index="1"/> because I think it is  
    dangerous.
```

```
  </template>
```

```
</category>
```

- The system uses Cyc's knowledge of things to express an opinion. In this case it doesn't play with items Cyc thinks are dangerous. This is a quick way of specifying agent responses based on Cyc's knowledge of things.

ASKING A D3 CHARACTER ABOUT DANGEROUS THINGS

Do you like Ebola?
I don't play with
Ebola because I
think it is
dangerous.



SEMANTIC PATTERNS USING DENOTATIONAL MAPPER AND GENLS

```
<category>
  <pattern>WHO IS THE * OF *</pattern>

  <guard>( <cycterm filter="BinaryPredicate" ><star index="1"/></cycterm>
    <space/>
    <cycterm><star index="2"/></cycterm> ?REPLY)
  </guard>

  <template>
    I believe
    <cycterm>(cyc-query '(
      <cycterm filter="BinaryPredicate" ><star index="1"/></cycterm> <space/>
      <cycterm><star index="2"/></cycterm> ?REPLY) #$EverythingPSC)
    </cycterm>
    defines the
    <cycterm filter="BinaryPredicate"><star index="1"/></cycterm>
    of
    <cycterm><star index="2"/></cycterm>.
  </template>
</category>
```


KB LOOKUP

- who is the **spouse** of Bill Clinton ?
- [I believe (((?REPLY . # \$HillaryClinton))) defines the # \$spouse of # \$BillClinton.]
- what is the **population** of Sweden ?
- [I believe (((?REPLY . 8865051))) people live in Sweden.]
- who is the **wife** of Bill Clinton ?
- [A 20th century celebrity of some sort?]
- who is the **mate** of Madonna ?
- [I believe (((?REPLY . # \$SeanPenn))) defines the # \$mate of # \$Madonna.]

CYN-D KEY POINTS

- Characters can utilize Cyc and AIML in their natural language processing
- Doom scripts and state machines can assert and test CycL conditions
- Game events and data are reflected in the ontology
- The ontology is used to generate and infer game information
- Character planning uses Cyc's representation of the world
- Multiple tools co-exist in the Java VM space utilizing both Cyc and Doom as needed
- Non-logical, pattern based processing is interacting with Cyc

BURC: BOOTSTRAPPING USING RESEARCHCYC

- Goal: To extend Cyc's knowledge base using *"relationships implied to be possible, normal or commonplace in the world"*
- Prior work with Cyc knowledge entry has been manually oriented
- How will we collect common sense without a body and manual labor...?
- Read, Parse, Mine!
- Proposal: Read text, Parse into a database, Extract relations between words, Propose hypothetical relations between concepts

WHAT IS COMMON KNOWLEDGE ?

- Using an information channel model:
 - Information the Sender considers the Receiver to already know
 - If the Sender **does** send the info then ...
 - Receiver will consider the **Sender** to 'lack intelligence or experience' (*the sender is stupid*)
 - Receiver will believe the sender thinks the **Receiver** 'lacks intelligence or experience' (*the sender thinks I'm stupid, and needs to tell me*)
 - Possibly the Sender is clarifying which among many possible common options they mean in this case
 - Since both parties know the information, to send it would generate **negative information content / wasted bandwidth**
- Explains why it is hard to find common sense on the Internet!

BURC: BASIC ANALOGY

- The Shotgun approach to the Human Genome
- Extract millions of fragments
- Knit them back together by finding commonalities
- Will it work for the Human Genome?
- James Burke: 'Mr. Connections'

BOOTSTRAPPING WITH RESEARCHCYC

- Cyc has vocabulary about objects in the world and relationships
- It can always use more !
- BURC uses what Cyc already has + lots of parsed text to create new Cyc entries for common relationships found in the text
- Lenat's Bootstrap Hypothesis: once Cyc reaches a certain level/scale it can help in its own development and start using NLP to augment its knowledge base
- BURC should help test this hypothesis

THE BURC PROCESS: FROM SEEDS... HYPOTHE-SEED'S

- Use the Rosetta link grammar parser for bulk parsing of text, primarily narratives based in 'worlds like ours'. Other text styles could be included
 - Link Grammar Parser developed at CMU
 - Daxtron extended LGP into multi-format Rosetta
 - Cycorp's new "cyclify" function is LGP based
- Operates in two directions:
 - Forward from text to CycL
 - Backwards from existing CycL to the text to find new forward patterns

BURC PROCESS - 2

- Load the link parse fragments into a database (1 and 2 link fragments), and compute frequency of fragment occurrences. The database is in SQL format so multiple queries can be formed dynamically. [Read Once/Use Many]
- Using Cyc knowledge as a starting point (the seeds), extract knowledge for use in Cyc:
 - Given a set of seed facts in Cyc, identify how those facts are represented as link fragments in the database
 - Generate conjectures as to new knowledge AND new knowledge extraction patterns using the fragment patterns

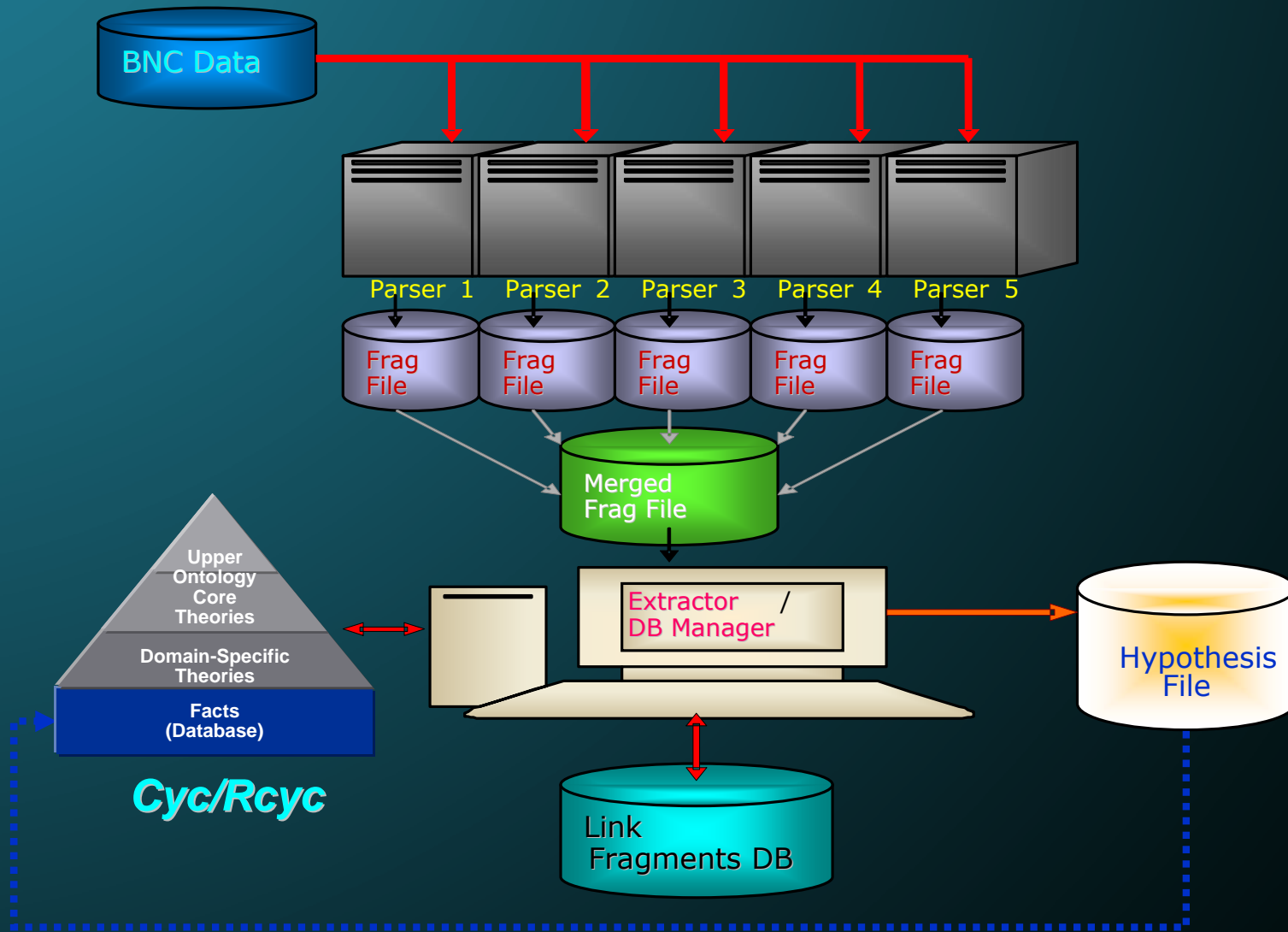
BURC PROCESS - 3

- Use Cyc knowledge directly to conjecture new statements:
 - Cyc has lexical knowledge, which can be used as templates against the DB to form new statements
 - For example, common adjectives applied to noun classes
 - Cyc knows “WhiteColor” and “Blouse” but does not know that white is a common blouse color, although it becomes apparent after reading some text
- Optionally, gather supporting background statistics for hypothesis verification using other sources:
 - Perhaps Google desktop with a larger than fully parsed corpus
 - Perhaps check against answer extraction engines

MINING ADJECTIVE KNOWLEDGE EXAMPLE

- “white blouse” as factoid fragment
- In Link Grammar format
[white.a | a | blouse.n]
- Potentially test using an internal or search engine relevancy metric [GC=70400]
- (adjSemTrans White-TheWord 11
RegularAdjFrame (mainColorOfObject
:NOUN WhiteColor))
- Hypothesis: (*plausibleValueType Blouse
mainColorOfObject WhiteColor*)

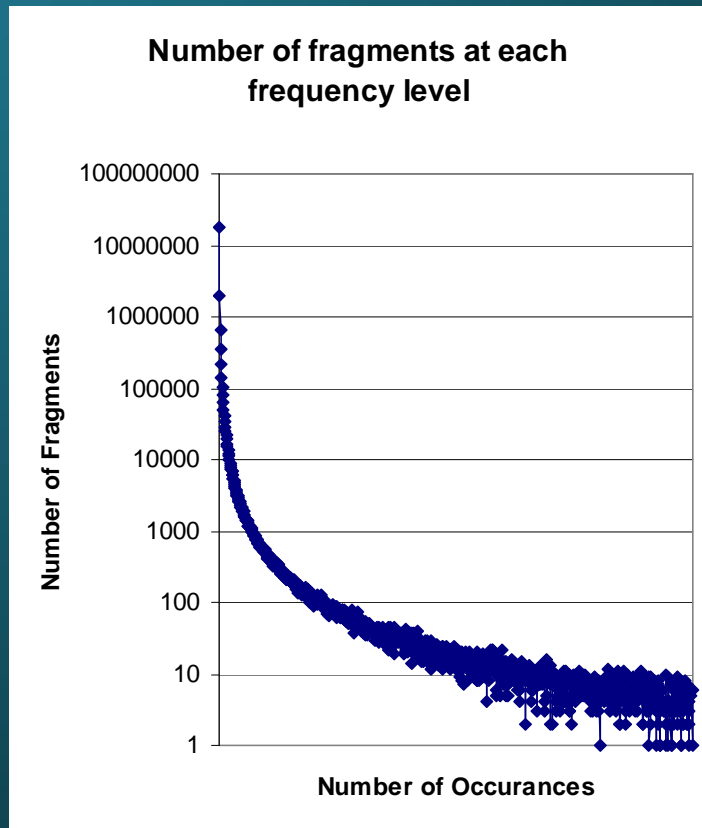
FLOW OF PROCESSING



RUNNING THE SYSTEM

- Used a filtered set of the BNC (650 Meg of data)
- 5 parsers running in parallel for 70 hours generated 1.91 Gig of output
- Reduced to 1 Gig of unique records with counts
- 783 Meg or 22 million fragments

FREQUENCY OF FRAGMENTS



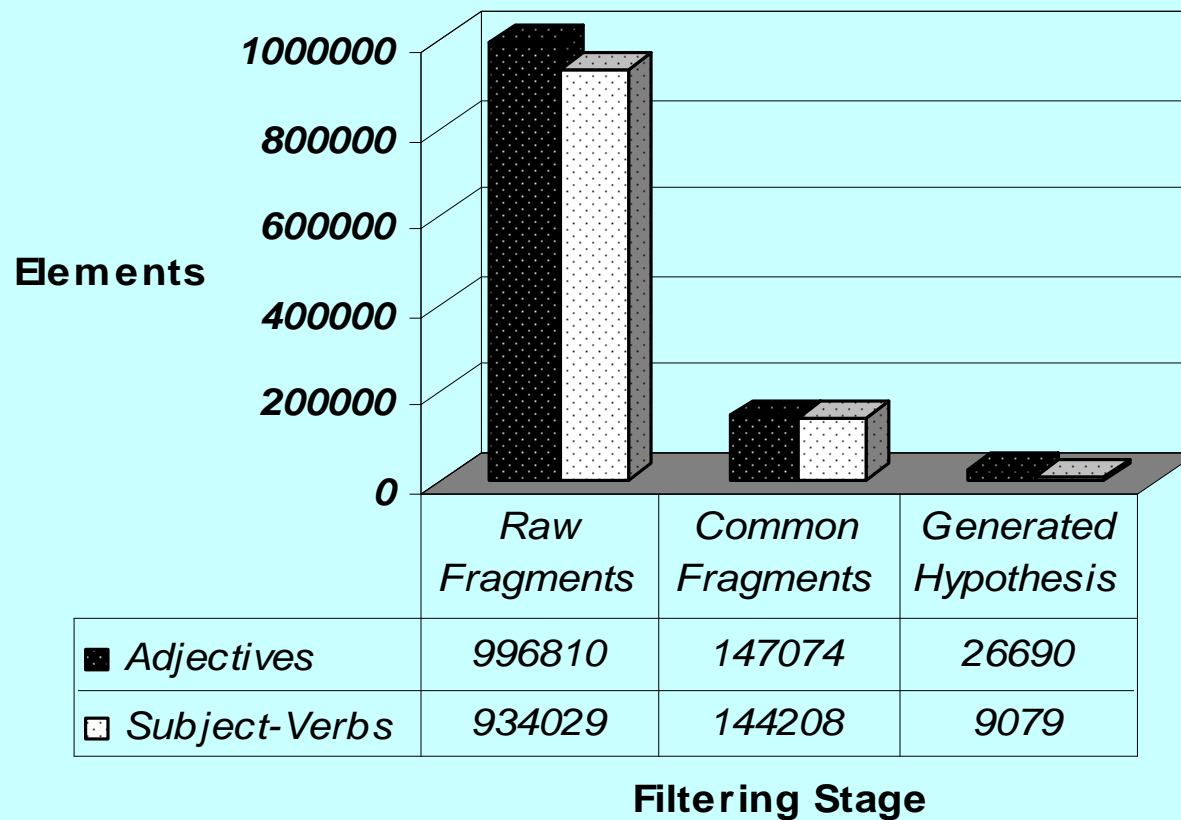
- The distribution of fragments follow a smooth curve in log space
- Similar to zipf distribution for words, characters and n-grams

THE HUNT FOR COMMON FRAGMENTS

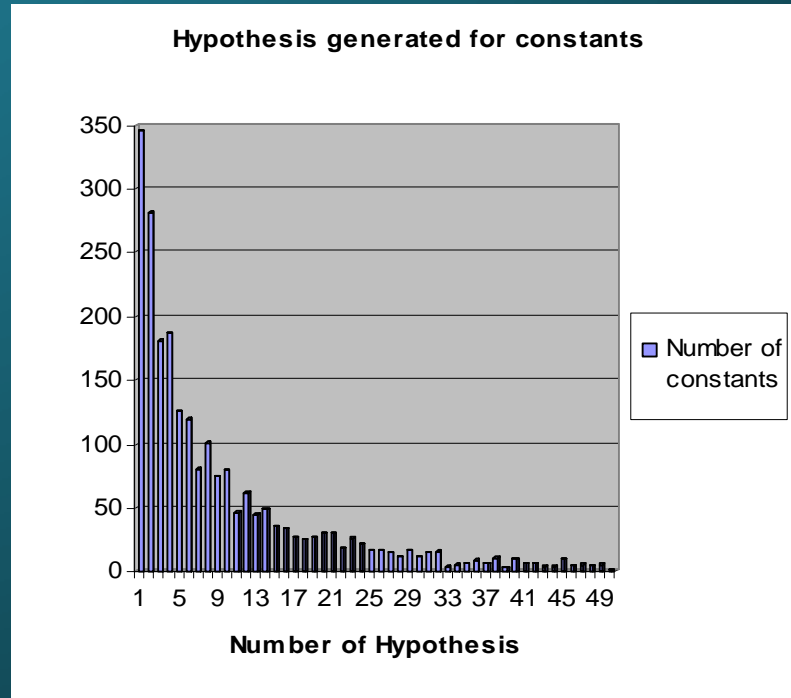
- Forward mining was run over adjective links with more than one fragment and subject-verb pairs with more than two fragments
- In both cases this was approximately the top 15% for each search class

DISTILLATION

From Fragments into Hypothesis



A SOURCE OF POTENTIAL KNOWLEDGE



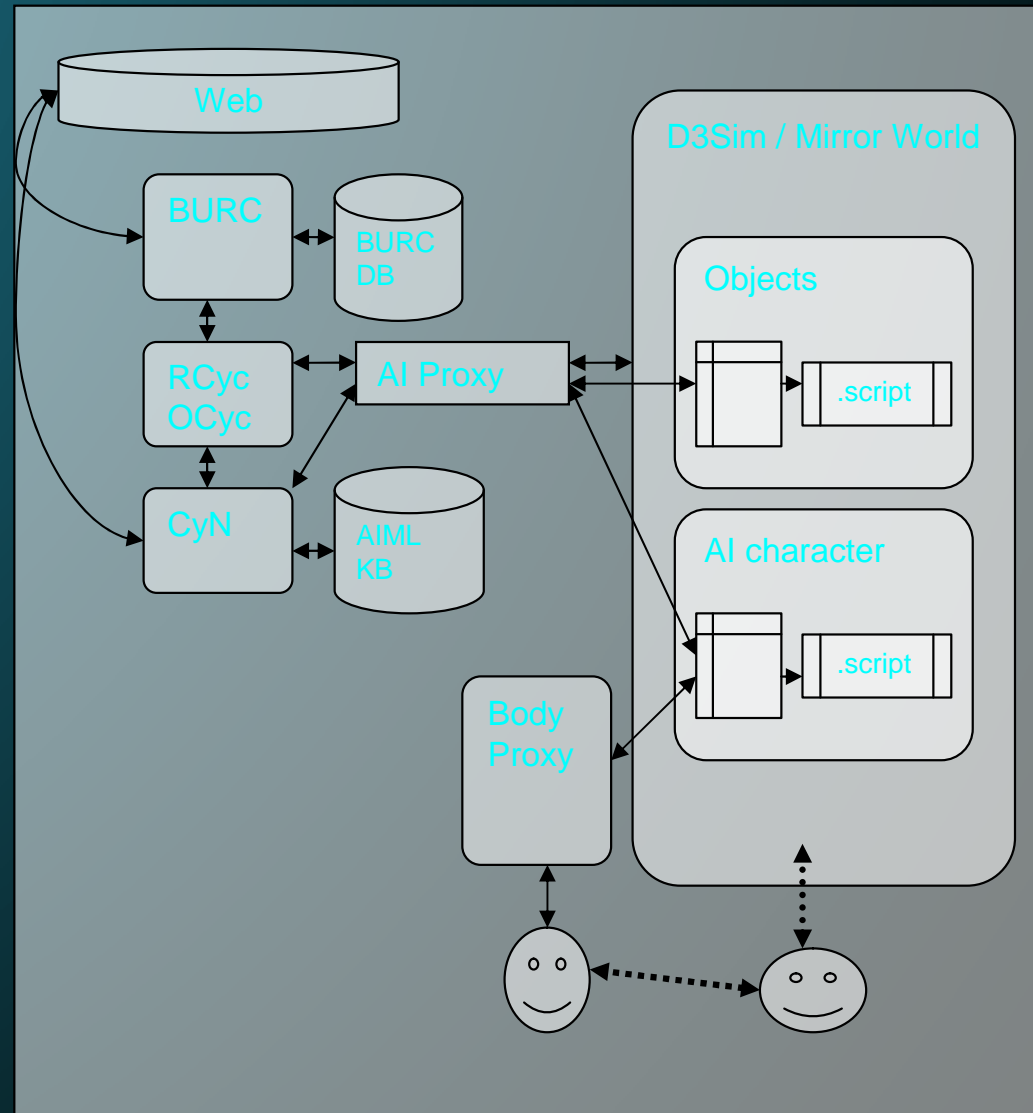
- The various versions of Cyc have 10 to 20 assertions per constant
- BURC generates 14.29 hypothetical assertions per constant
- Need to quantify the quality of BURC knowledge

CAN WE EVER BE “DONE”?

- Need to explore the definition of semantic coverage metrics for unmapped domains
 - The space of 2.4K of binary predicates applied to 85K constants provides a 16 trillion combination search space, only a fraction of would be considered part of ‘common knowledge’
 - It is a big space, but finite and within range of current technology
- Need to examine effect of reporting the unusual instead of the usual
- Need to utilize knowledge and experience on the backwards miner
- *Cyc can* help in its own knowledge entry process. 62% of subject-verb hypotheses were filtered out using semantic filtering

BOOM: HOW DOES IT ALL FIT?

From the web common sense is mined using knowledge extraction with BURC and verified with simulation and interaction using CyN-D



OVERALL BENEFITS - 1

- BURC lets Cyc do what people always wanted: have the AI expand its knowledge by reading texts
- CyN-D provides a concrete cross-domain training environment
 - Opens Cyc to the simulation, gaming and chatbot programmer communities
 - Provides Cyc users with chatbots and 3D engines
- Cyn-D should help answer: Is this a good interface model?
- The Big Language Test: Can more people describe and solve more problems more easily with computers?
- Isn't this what Turing was really after?

OVERALL BENEFITS - 2

- Moves towards the goal of “Holodeck characters” as an interface
- Expands Cyc’s representation and interface capabilities
 - Representing the Doom 3 environment in Cyc
 - Spatial and other planning can be simulated in the Doom 3 engine
 - Potential for Cyc to “show” the user the results of its logic
 - Dealing with a ‘messy’ environment
- Provides Cyc with personality
- Provides chatbots and characters with first-order logic

QUESTIONS & CONTACT

- Daxtron Laboratories
 - <http://www.daxtron.com>
 - Kino Coursey
 - Kino@daxtron.com
 - Douglas Miles
 - Dmiles@daxtron.com
- University of North Texas
 - Dr Rada Mihalcea
 - <http://www.cs.unt.edu/~rada/>
 - rada@cs.unt.edu
 - Dr Paul Tarau
 - <http://www.cs.unt.edu/~tarau/>
 - tarau@cs.unt.edu