EE 542

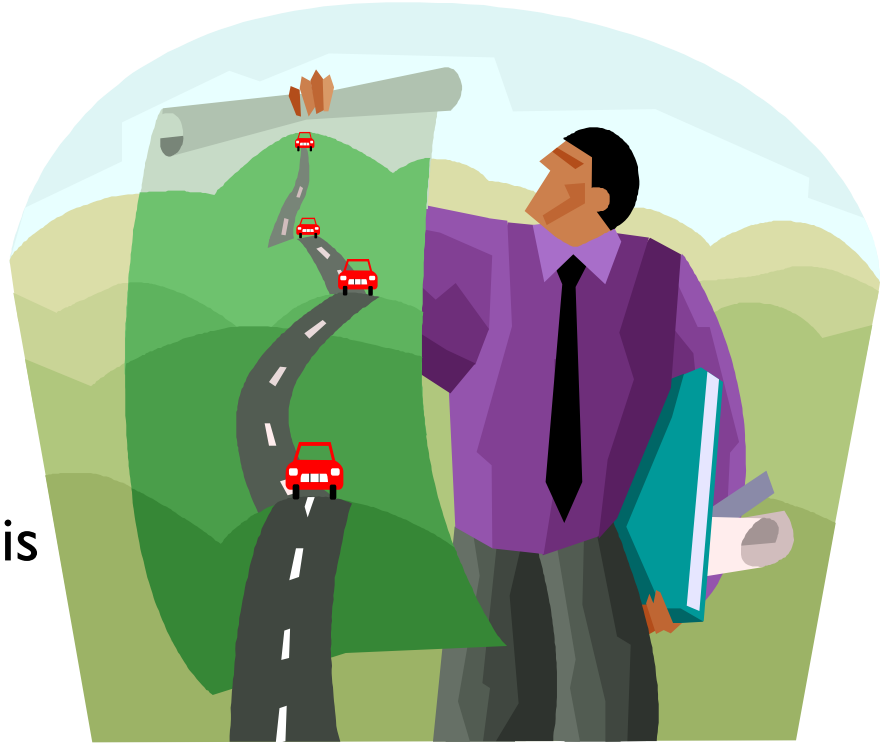
Lecture 16: Big Data Processing & Powerpoint Animations

Internet and Cloud Computing

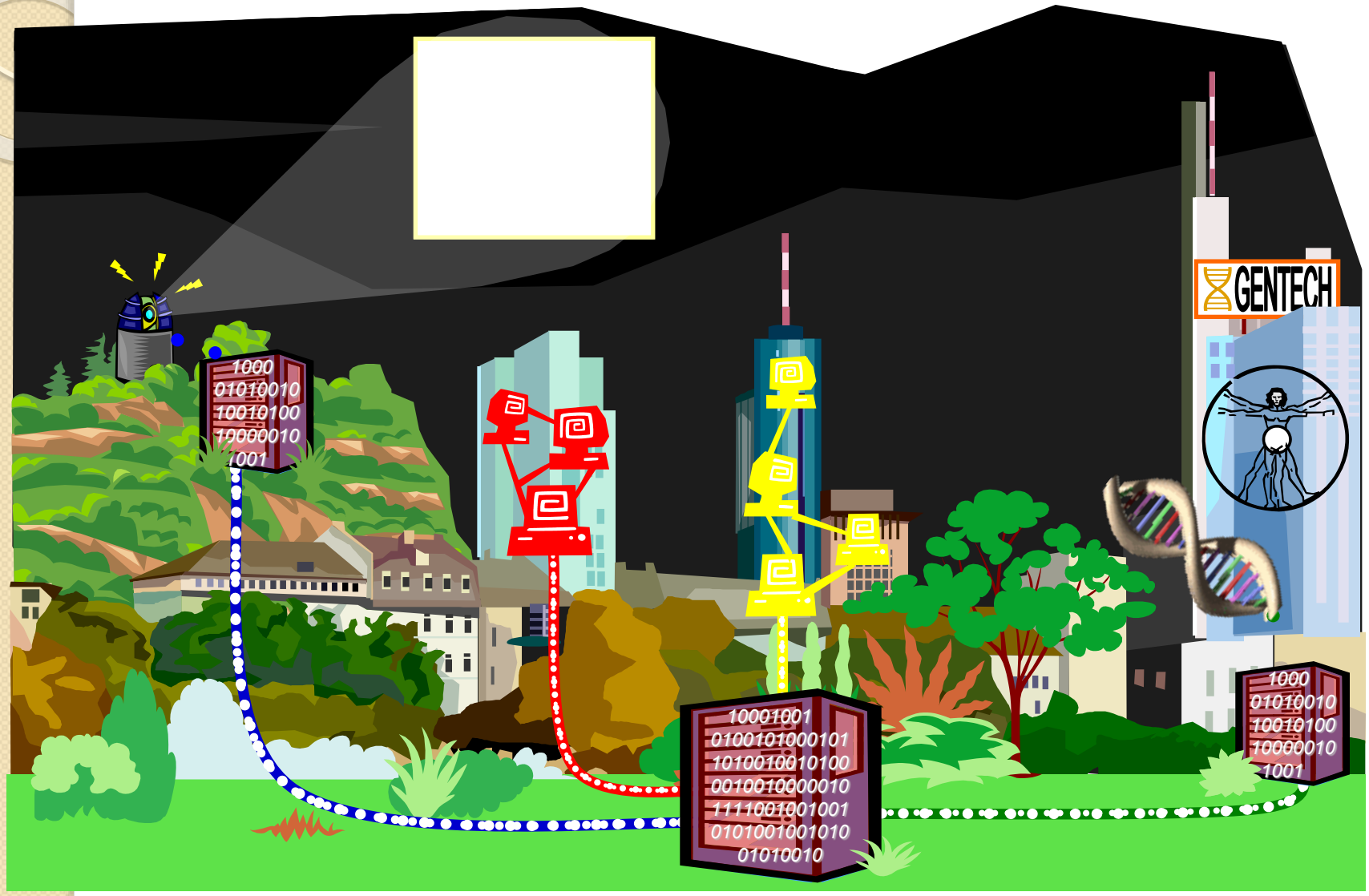
Young Cho
Department of Electrical Engineering
University of Southern California

Outline

- Massive Streaming Data
- Computer Networks
- Network Security
- Internet Document Analysis
- Research Directions

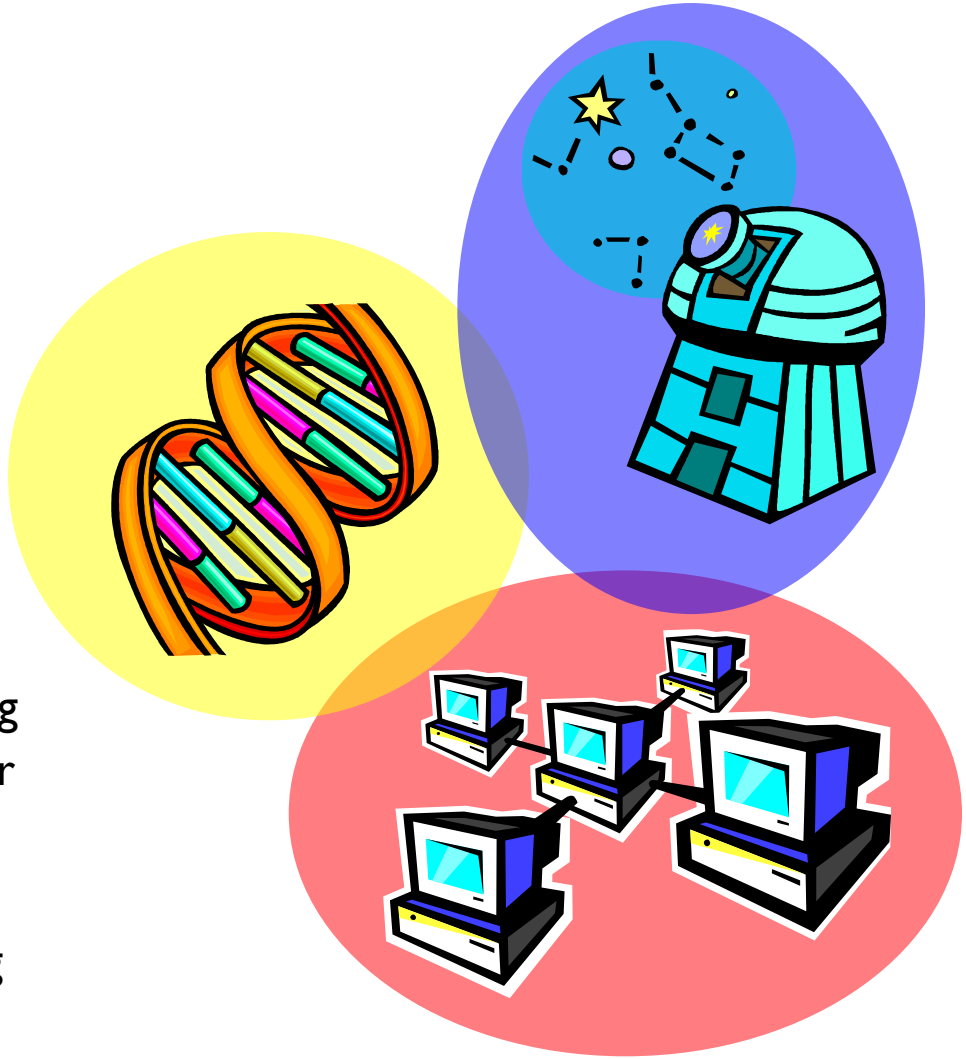


Massive Amount of Data



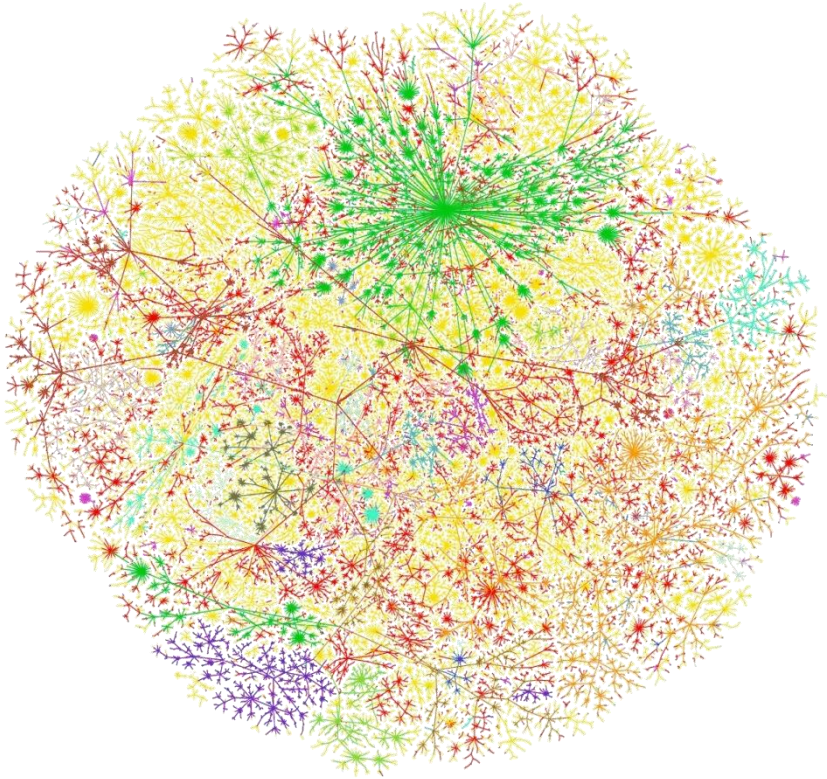
Massive Amount of Data

- Image and Signal Processing
  - 3-D Sonar Beamforming
  - Automatic Target Recognition
  - Video Stream Compression
- Pattern and Syntax Detection
  - Network Intrusion Detection
  - Network Data Extraction
  - Biosequence Parser
- Semantic Data Processing
  - Network Data Classification
  - Network Information Clustering
  - Content based Network Router
- High Performance Computing
  - High Performance Networks
  - Scalable Distributed Computing
  - Automatic Thermoregulation



Computer Networks

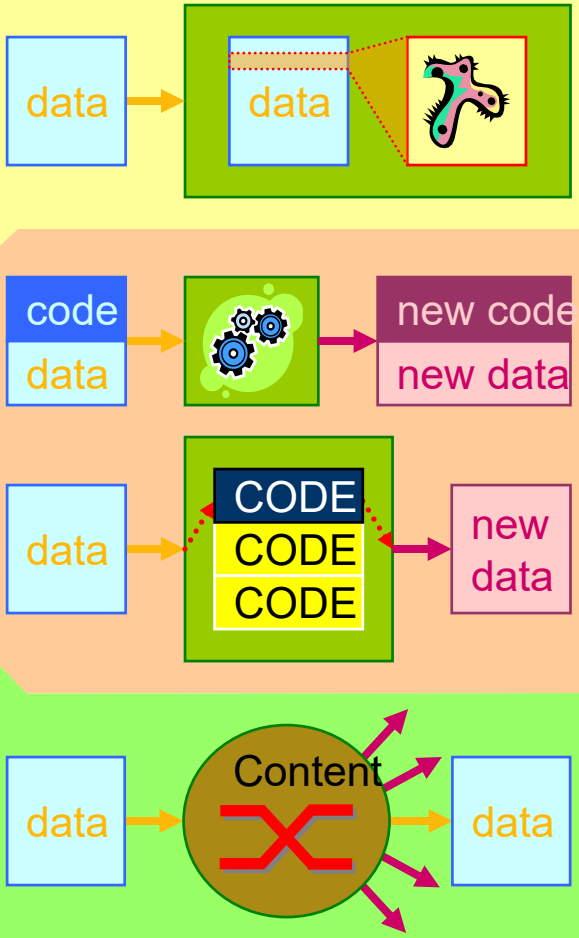
- Challenges
  - High Bandwidth
  - Continuous Streams
  - Insufficient Processing
- Current Practice
  - No Monitoring
  - Superficial Monitoring
  - Data Sampling
  - Batch mode Process



A Map of Internet in June 28, 1999

Processing Network Data

- Traffic Monitor
  - Packet Detection and Filtering
  - Packet Classification/Clustering
  - Network Forensics
- Processing Node
  - Packet Modifications
  - Data Extractions
  - Active Networks
- Packet Router
  - Header Field based Routing
  - Payload Content based Routing



Several Open Problems

- Network Security
  - Complex Attacks
  - 24/7 Attacks from Remote Locations
  - Huge Financial Loss
- Internet Data Analysis
  - Terrorism
  - E-commerce
  - Social Network

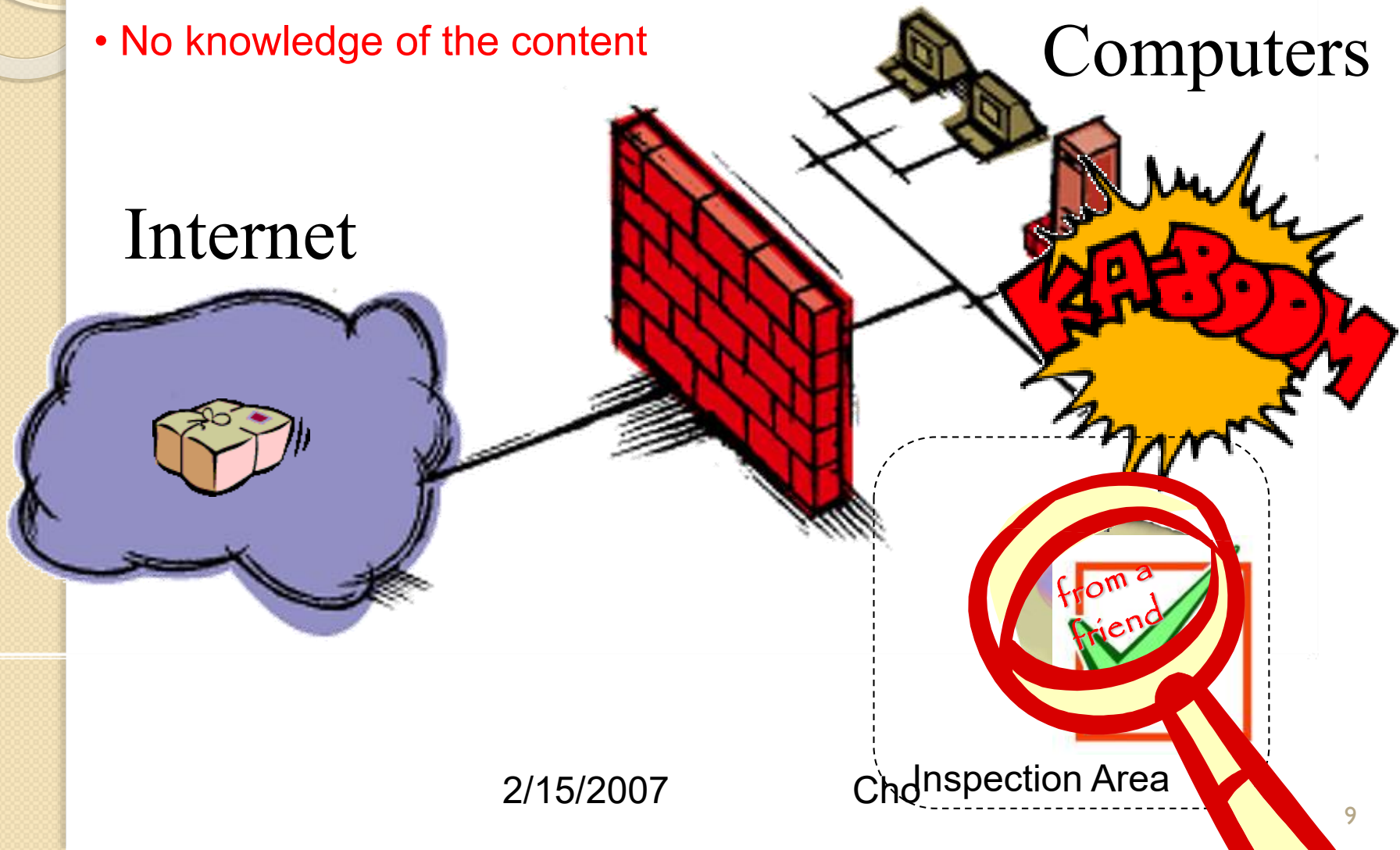
General Approach

- Clean Up Meta Data
- Abstract Based on Known Model
- Pattern Search
- Regression – Noise Filtering

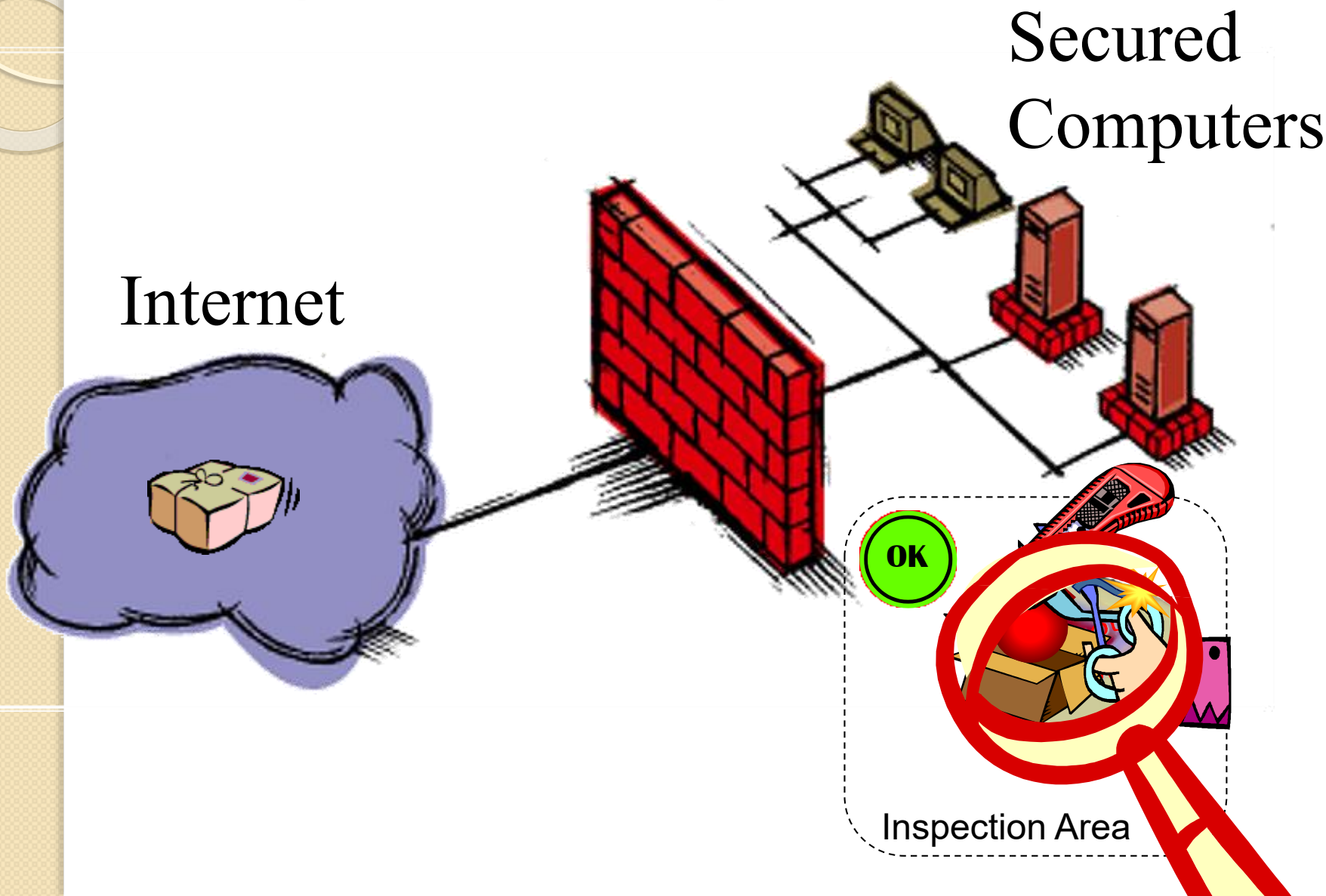
Detecting Network Intrusions

Minimal Processing: Scan the address

- No knowledge of the content

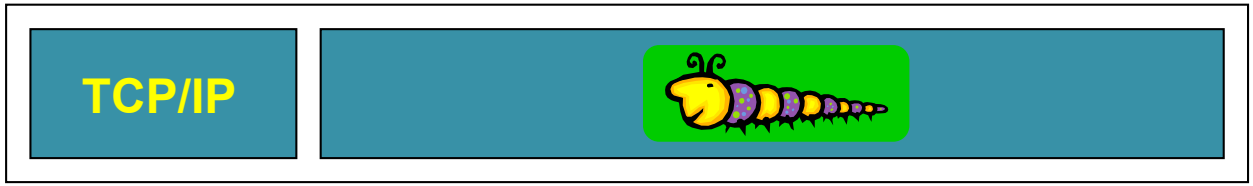


Deep Packet Inspection



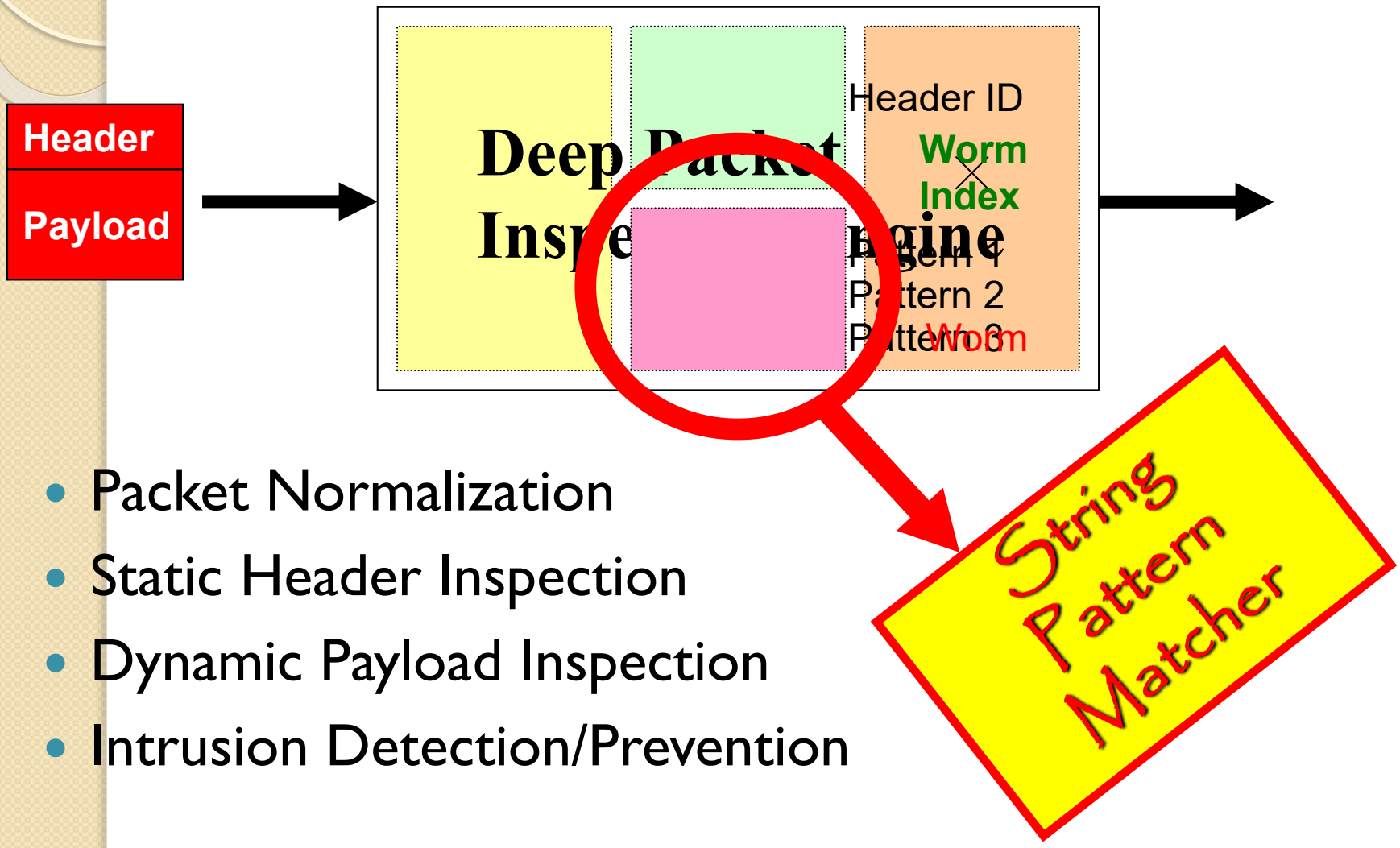
Characteristic of Attacks

- Static packet header information
- Attacks are embedded in packet payload
- Unpredictable location of attack pattern
- Solution: Deep Packet Inspection



Computer Network Packet

Deep Packet Inspection



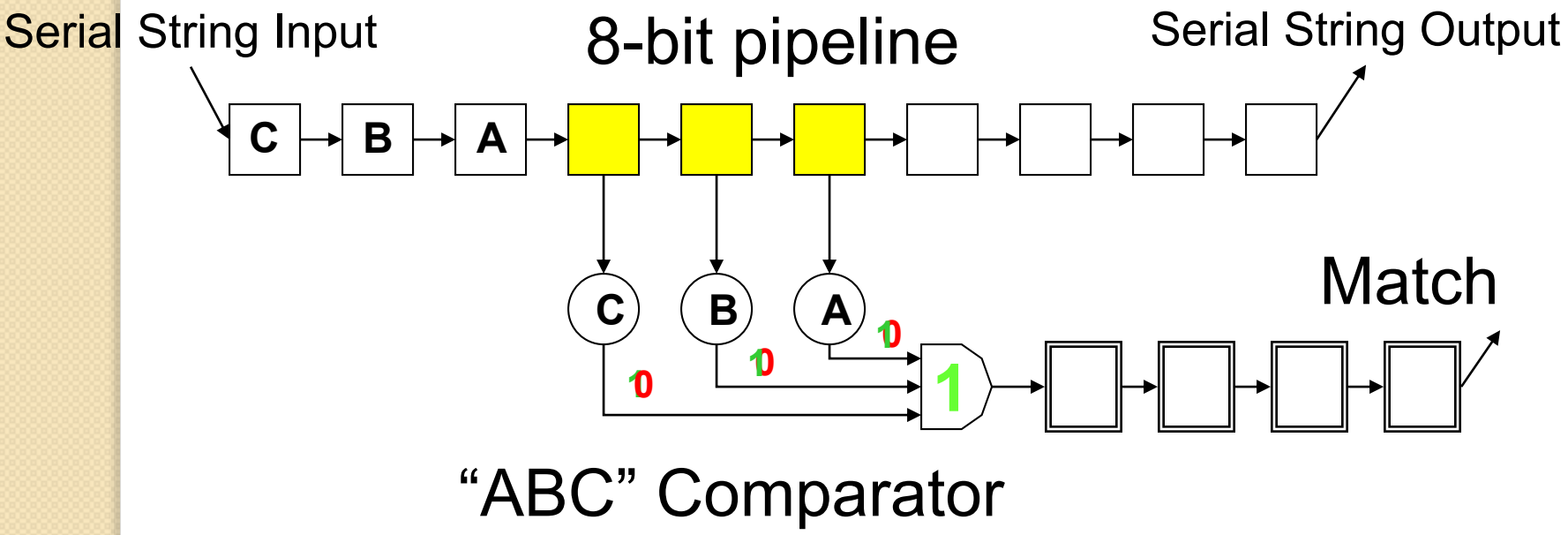
- Packet Normalization
- Static Header Inspection
- Dynamic Payload Inspection
- Intrusion Detection/Prevention

Design Platform

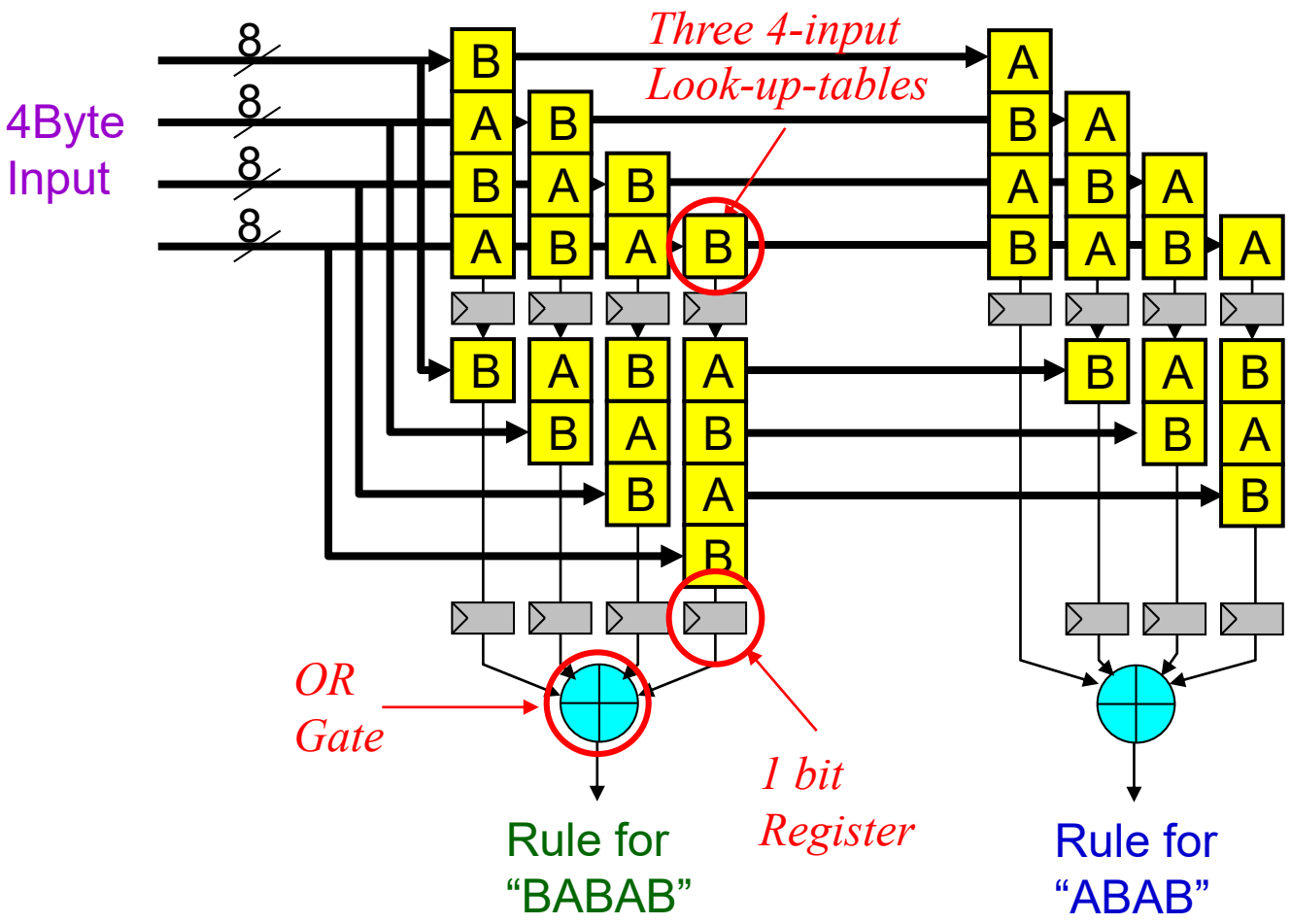
- General Purpose Processors
  - **Pro**: Flexibility of Software
  - **Con**: Limited Processing Performance
- Application Specific Processors
  - **Pro**: Higher Performance per Cost
  - **Con**: Limited Programmability
- Hardware Accelerated Devices
  - Leverage Parallel Hardware Designs
  - Programmable Hardware
- **Cloud Computers?**

Pattern Matching in FPGA

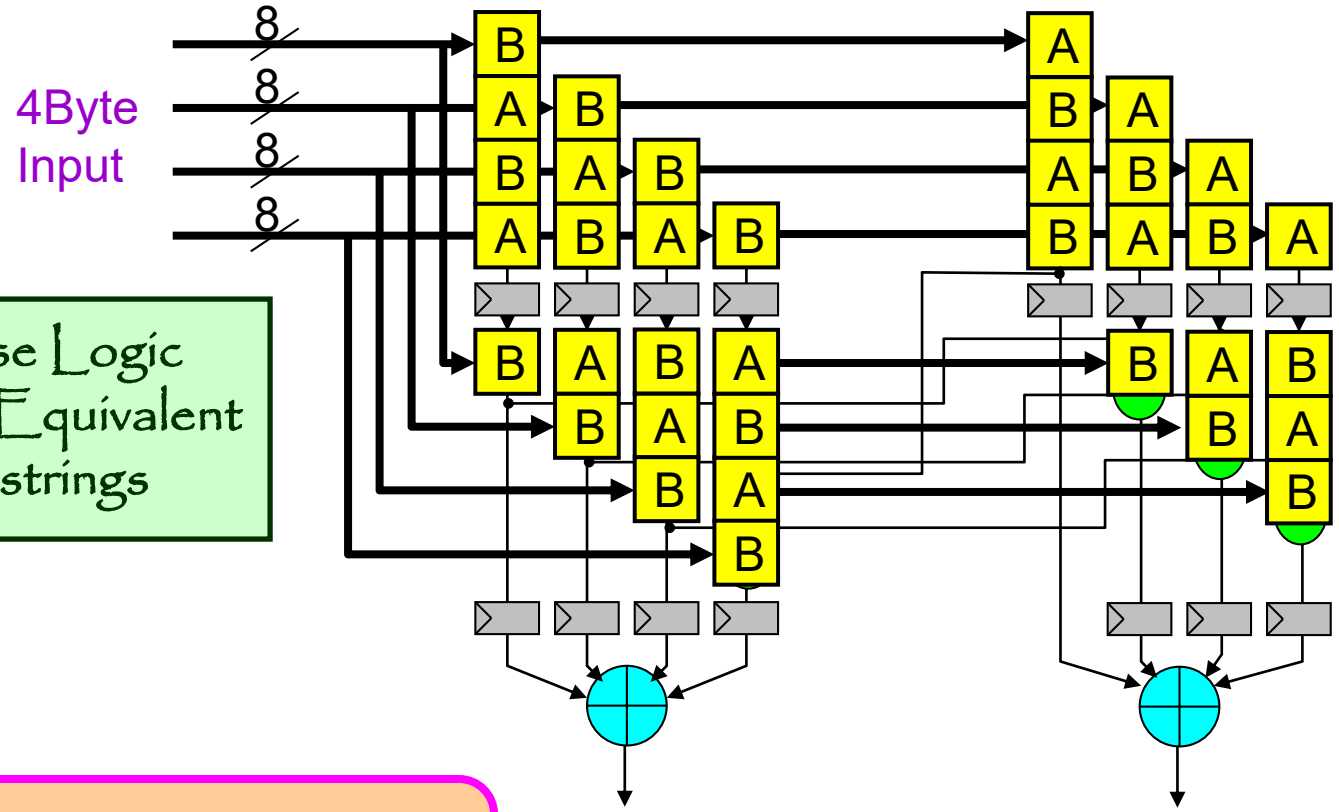
Matching "ABC" in serialized string



Scalable Pattern Matcher



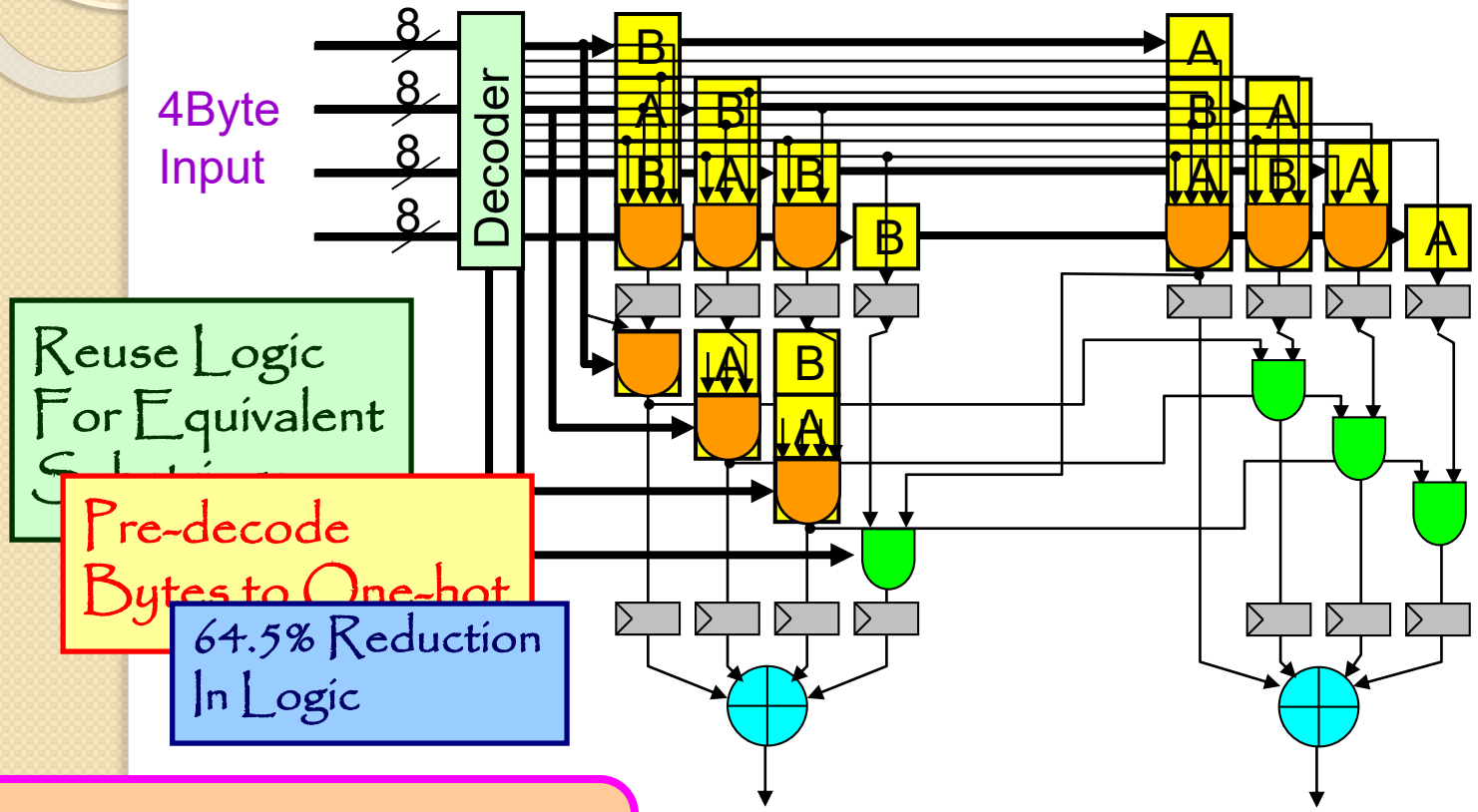
Scalable Pattern Matcher



Reuse Logic For Equivalent Substrings

Number of 4-input LUTs = ~~8~~ 10

Scalable Pattern Matcher



Number of 4-input LUTs = 39

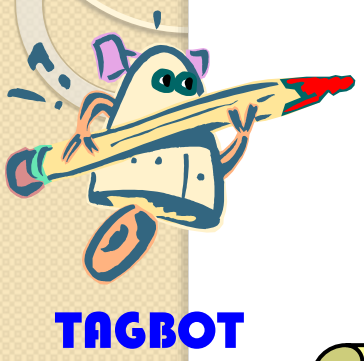
New Infrastructure/New Methods

- Cloud Computing Platforms?
- How Would You Do This?
- What Platform Would You Use?
- Approach
  - Pre-filtering
  - Parallel Processing – Data Differentiation
  - Deep Data Analysis – Final Verdict

Big Data

- Methodology
  - String Pattern Matchers
  - Network Intrusion Detection System
  - Bag of Words based Document Classification
- Towards Advanced Data Processing
  - Detecting Syntax and Semantics of data
  - Using grammatical structure to process data
  - Data structure with probabilistic models

Meta Data Filtering

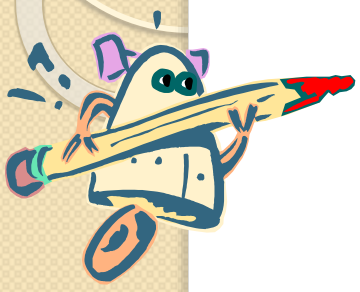


HTML Source Document

```
<h2>Company Overview</h2>
Corporate Fact Sheet</a></b>Founded by
"/about/profile.html">Dr. J. Robert Beyster</a> and a
small group of scientists in 1969, SAIC, a Fortune 500
company, now ranks … and have more than 43,000 <!-- Also
update employee number on: saic.com/news/0722.html -->
employees with offices in over 150 cities.
```

- String Patterns
  - HTML tags can be detected and marked
  - Marks can be used to filter out the tags
  - Discrete gates, Memory based, Hybrid filter, Bloom filter and etc.
- Detect and Filter out HTML tags
  - <h2>, </h2>, , , <a href=, , <!--, -->
- Some unwanted texts are still not filtered away!

Semantic Analysis



HTML Source Document

```
<h2>Company Overview</h2>
<!-- Corporate Fact Sheet --> <p>Founded by <a href=
"/about/profile.html">Dr. J. Robert Beyster</a> and a
small group of scientists in 1969, SAIC, a Fortune 500
company, now ranks … and have more than 43,000 <!- Also
update employee number on: saic.com/news/0722.html -->
employees with offices in over 150 cities. </p>
```

Token List

(1) hdr2 : 'h2'
(2) para : 'p'
(3) link : 'a'
(4) href : 'href='
(5) quot : '"'.alpnum*.'"'
(6) comm : alpnum*
(7) strg : alpnum*

Simple HTML Grammar

(1) Tag_Name → hdr2 | para | link
(2) Comment → '<!-'.comm.'-->'
(3) Attrib → href.quot | ϵ
(4) Tag_Head → '<'.Tag_Name.Attrib.'>'
(5) Tag_Tail → '</'.Tag_Name.'>'
(6) Expr → Comment | strg | ϵ
(7) Line → Tag_Head.Line.Tag_Tail
| Expr.Line.Expr | Expr
(8) Content → Line.Content

Language Parsing

Grammar

```
STRING [a-zA-Z0-9-]+
%%
card:        "<card>" rout
"</card>"
routekey:    "<routekey>"
route:        routefirst
routefirst:  "first"
routelast:   "last"
name:        "<name>" name
nameN:        nameFL | nam
nameFL:       firstFL last
nameLF:       lastLF first
firstFL:     "<first>" STR
lastFL:      "<last>" STRI
lastLF:      "<last>" STRI
firstLF:     "<first>" STR
title:       "<title>" STR
phone:       "<phone>" STR
%%
```

Tokens

```
0)  <card>
1)  </card>
2)  <routekey>
3)  </routekey>
4)  first
5)  last
6)  <name>
7)  </name>
8)  <first>
9)  [a-zA-Z0-9-]+
10) </first>
11) <last>
12) [a-zA-Z0-9-]+
13) </last>
14) <title>
15) [a-zA-Z0-9-]+
16) </title>
17) <phone>
18) [a-zA-Z0-9-]+
19) </phone>
```

Grammar

Tokenizer

```
STRING [a-zA-Z0-9-]+
%%
card:        "<card>" routekey name title phone
"</card>"
routekey:    "<routekey>" route "</routekey>"
route:        routefirst | routelast
routefirst:  "first"
routelast:   "last"
name:        "<name>" nameN "</name>"
nameN:        nameFL | nameLF
nameFL:       firstFL lastFL
nameLF:       lastLF firstLF
firstFL:     "<first>" STRING "</first>"
lastFL:      "<last>" STRING "</last>"
lastLF:      "<last>" STRING "</last>"
firstLF:     "<first>" STRING "</first>"
title:       "<title>" STRING "</title>" | ϵ
phone:       "<phone>" STRING "</phone>" | ϵ
%%
```

Token Bits

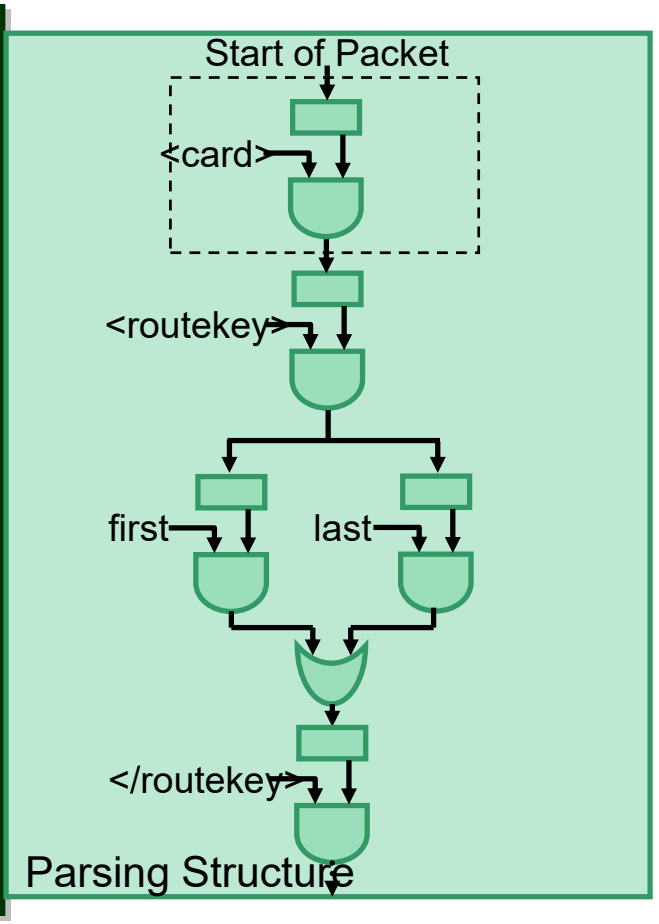
Tokens

```
0)  <card>
1)  </card>
2)  <routekey>
3)  </routekey>
4)  first
5)  last
6)  <name>
7)  </name>
8)  <first>
9)  [a-zA-Z0-9-]+
10) </first>
11) <last>
12) [a-zA-Z0-9-]+
13) </last>
14) <title>
15) [a-zA-Z0-9-]+
16) </title>
17) <phone>
18) [a-zA-Z0-9-]+
19) </phone>
```

Grammar Parser

Grammar

```
STRING [a-zA-Z0-9-]+
%%
card:         "<card>" routekey name title phone
"</card>"
routekey:     "<routekey>" route "</routekey>"
route:         routefirst | routelast
routefirst:   "first"
routelast:    "last"
name:         "<name>" nameN "</name>"
nameN:         nameFL | nameLF
nameFL:        firstFL lastFL
nameLF:        lastLF firstLF
firstFL:      "<first>" STRING "</first>"
lastFL:       "<last>" STRING "</last>"
lastLF:       "<last>" STRING "</last>"
firstLF:      "<first>" STRING "</first>"
title:        "<title>" STRING "</title>" | ϵ
phone:        "<phone>" STRING "</phone>" | ϵ
%%
```



Processing E-mail

```
From sender@bang.smtp.server.com Fri Jul  7 13:16:33 2006 -0500
Return-Path: <sender@smtp.server.com>
X-Original-To: sender@smtp.server.com
Delivered-To: sender@smtp.server.com
```

Remove Email Headers and HTML Tags

Convert Text

```
ICi0ur/lPb+sx9W0ur26KSDAzLehv+4gxq/GxL/4ID0gx9G8urfEIMCvv6PB1sDnILrPx9G068el
us4gwve8rrTru+e0wiA3wM8ox/bB9r3DsKMpICK5zLG5wMwguLbEq7/AILnmxNq1qMW4vsa9w77G
KEJEQSkgwLrH4MDHCrW/sOHA2rHdwLsgx6646SA2wNrIuLTjv6EgurmxzcfSILz2IMDWwLsgsM0i
wMy287DtILi7x9+02S4KCsfRIML3vK6067vntMIgwMyzryC/rMfVtLq9ur/NIMD8yK3F68itv6G8
rSAiucyxucDMIL/suK6/zSCw+MG4x9IgwMe757ChIMDWtMCzxCwgtscgua7BprimILTryK24psXr
x9ggx64gvPawoSDA1rTAs8SwoSDB37/kx8/B9iC068itwMcKx/y9xMC6IMHfv+THz8H2IL7KtNki
uOm8rSAiuLbEq7/
```

(고뉴스=김성덕 기자) 아프카니스탄 바그람 기지 앞에서 폭탄테러에 숨진 다산 부대고 윤장호 병장의 비보가 전해지면서 해외파병 부대의 안전에 대한 우려가 높아지고 있는 가운데, 다산부대 복무경험이 있는 전역병이 아프카니스탄의 근무여건과 상황을 털어놨다.

지난 2004년 8월부터 2005년 2월 중순까지 다산부대에서 통역병으로 근무한 천영록 씨는 "환경적으로는 거의우리나라 사람은 접해보기 힘들 정도로 열악한 곳"이라며 "바그람 고지는 모래투성이인데, 모래가 우리가 생각하는 것처럼 휘날리는 모래가 아니라 딱딱하고 건조한 땅이고,모래 바람 같은게 한 번 들어오면 바람때문에 얼굴이 아플 정도의 강풍이 이틀의 한 번, 하루의 한 번 세게 불어오고 그랬었다"고 회상했다. Outbound message

천 씨는 28일 CBS '김현정의 이슈와 사람'과의 인터뷰에서 "(윤병장의 죽음이) 정말 안타까운 일이지만 사실 가 있던 입장에서 보면 그 위험이 그 당시에도 상당히 많았었고, 파병을 갈 때도 상당부분 각오를 하고 가는 부분이기도 했다"며 "위험은 계속 주위에 있었는데, 저희 통제 밖이어서 굉장히 위험한 곳이지만 저희가 어떻게 해볼수 없는, 그냥 각오만 하고 산 경우"라고 당시의 심경을 고백했다

87/Thu Jul 6 15:55:48 2006
com

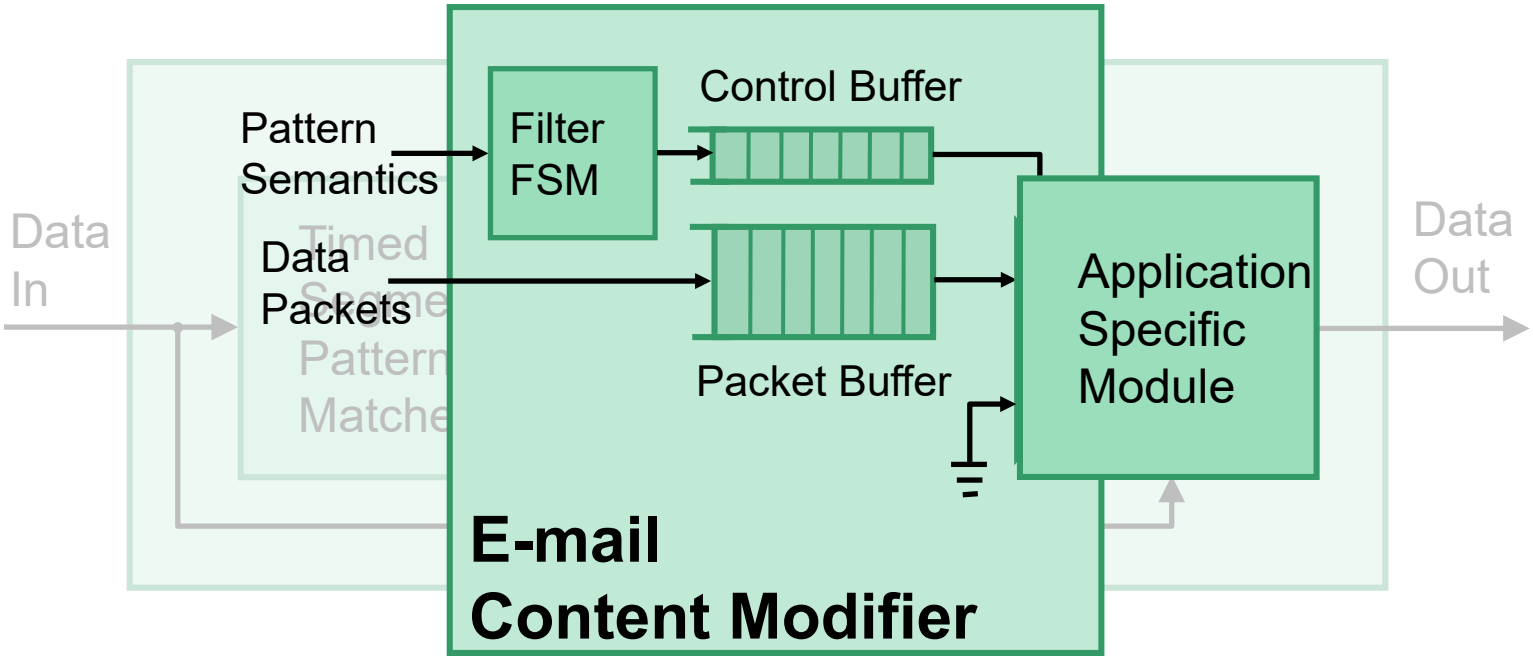
```
!=-4.6 required=6.0 tests=ALL_TRUSTED,AWL,BAYES_00,
            HTML_40_50,HTML_MESSAGE autolearn=ham version=3.0.5
X-Original-Status: O
X-Original-X-UID: 3972
Content-Length: 5040
X-Original-X-Keywords:
------=_NextPart_000_000C_01C6A1C7.9102C700
Content-Type: text/html;
            charset="utf-8"
Content-Transfer-Encoding: quoted-printable
```

Extracting and Conversion of E-mail Body

RFC-2822

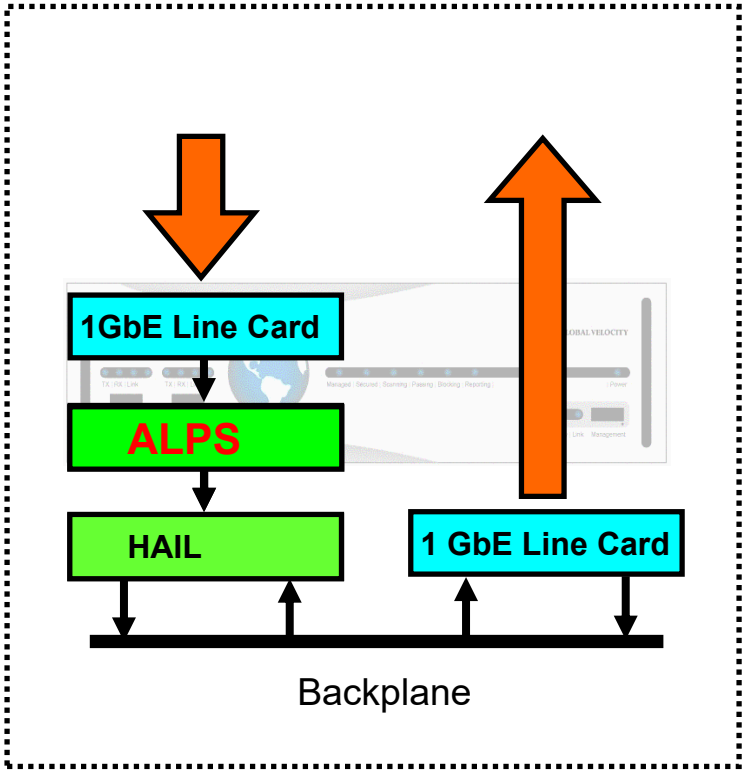
```
;
; RFC 2822
;
ALPHA =  %x41-5A / %x61-7A   ; A-Z / a-z
BIT =  "0" / "1"
CHAR =  %x01-7F
CR =  %x0D
CRLF =  CR LF
CTL =  %x00-1F / %x7F
DIGIT =  %x30-39
DQUOTE =  %x22
HEXDIG =  DIGIT / "A" / "B" / "C" / "D" / "E" / "F"
HTAB =  %x09
LF =  %x0A
LWSP =  *(WSP / CRLF WSP)
OCTET =  %x00-FF
SP =  %x20
VCHAR =  %x21-7E
WSP =  SP / HTAB
NO-WS-CTL = %d1-8 / %d11 / %d12 / %d14-31 / %d127
text = %d1-9 / %d11 / %d12 / %d14-127
specials = "(" / ")" / "<" / ">" / "[" / "]" / ":" / ";" / "@" / "\" / "," / "." / DQUOTE

quoted-pair = "\" text

FWS = [*WSP CRLF] 1*WSP
ctext = NO-WS-CTL / %d33-39 / %d42-91 / %d93-126
ccontent = ctext / quoted-pair / comment
comment = "(" *([FWS] ccontent) [FWS] ")"
CFWS = *([FWS] comment) (([FWS] comment) / FWS)

atext = ALPHA / DIGIT / "!" / "#" / "$" / "%" / "&" / "'" / "*" / "+" / "-" / "/" / "=" / "?" / "^" /
        "_" / "`" / "{" / "|" / "}" / "~"
atom = [CFWS] 1*atext [CFWS]
```

E-mail Packet Modifier



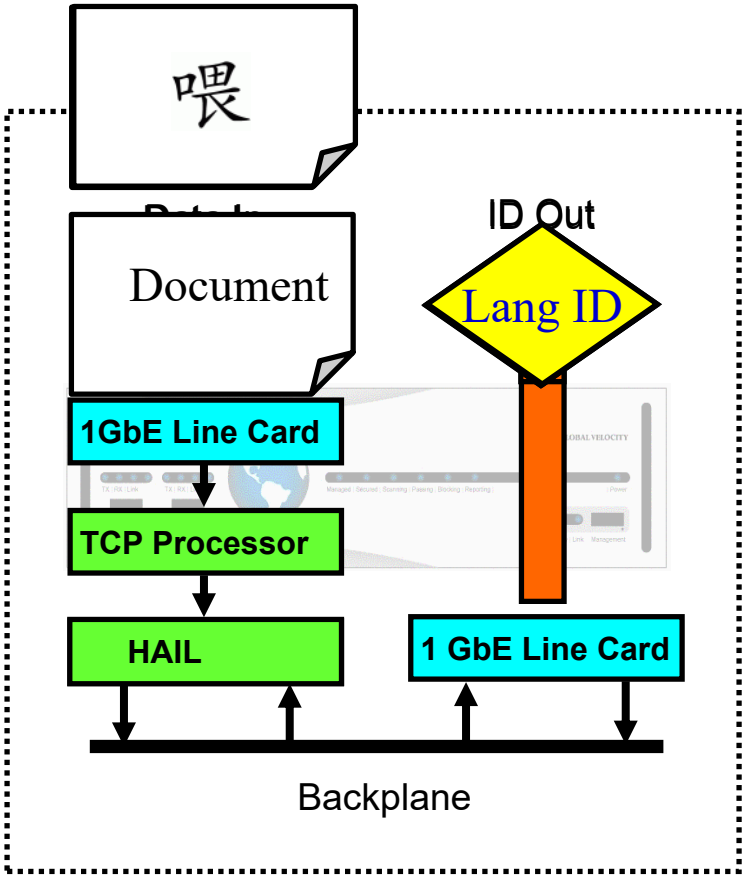
Technology Translation

- Method
  - Define Grammar for Input
  - Example: E-mail
  - Parse the data at Real-time
  - Send clean data to HAIL
- ALPS
  - Apply on flow
  - Tokenize data
  - Construct data structure
  - Filter out data
  - 10+ Gigabits/second



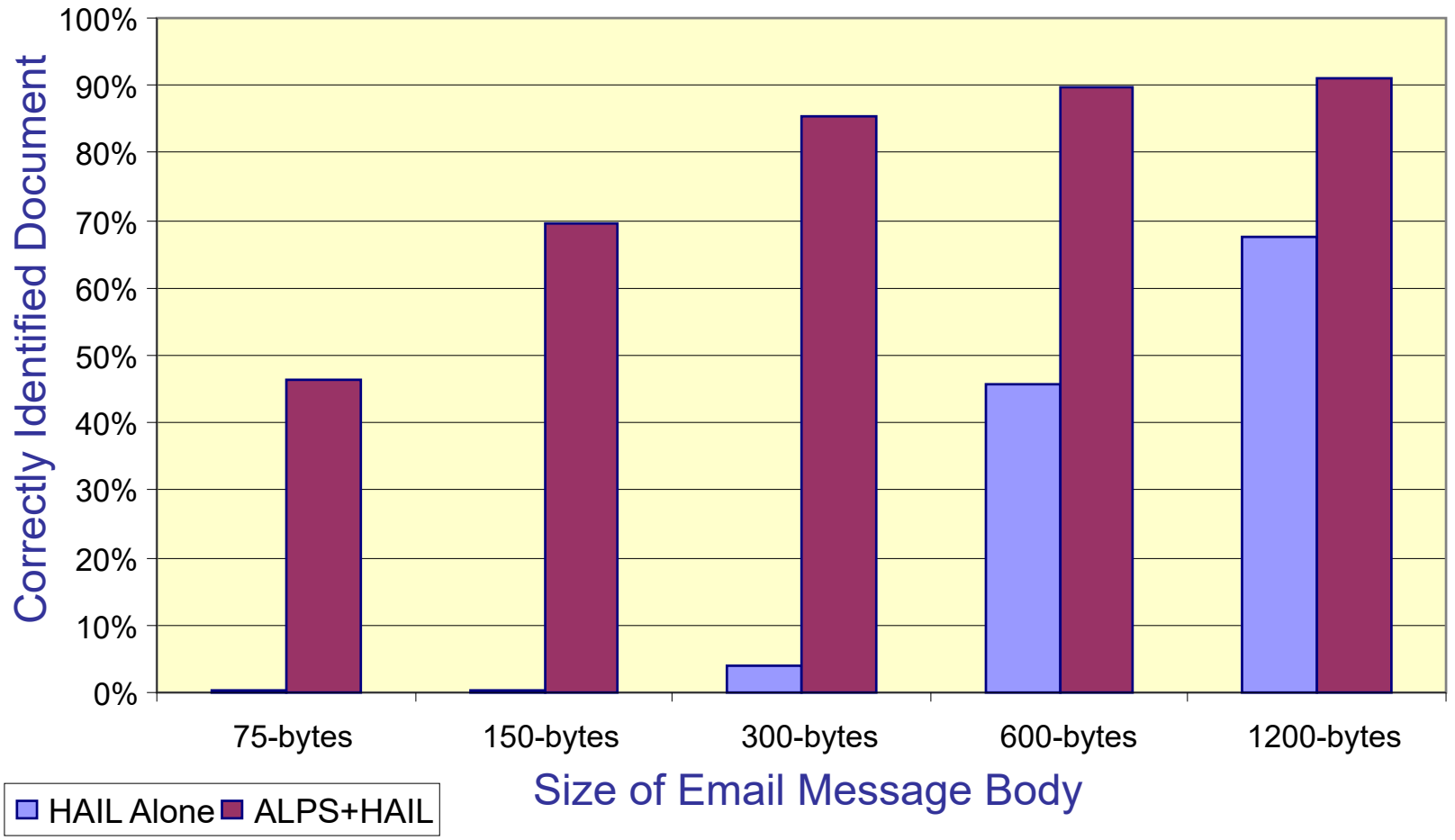
Identification of Languages

- HAIL
  - Above 90% Accuracy with Large Raw Documents
  - Can ID each packet
  - 2.4 Gigabits/second
- Platform
  - Field Programmable Port Extenders
  - Modular and Stackable Cards
  - Reconfigurable Devices
- Problem
  - Input documents are NOT clean
  - Packets can be small
  - HTML tags and E-mail headers
  - Variety of other attributes



Accuracy Improvement

10,816 email messages per data set in 14 different language documents



Computer Networks Research

- Data Processing
  - Grammar/Parser based Interpretation
  - Customizable Packet Fields and Actions
  - Hardware Accelerated System Response
- Platform Integration
  - Software based System Management
  - Hardware based Data Processor
  - Automated Hardware Generation
  - Cached bitfiles for Dynamic Reconfiguration
  - Micro/Macro Level Partial Reconfiguration
- 100+ Gbps Network Applications
  - Hardware Accelerated Publish/Subscribe Network
  - Reconfigurable Multi-protocol Router

Summary

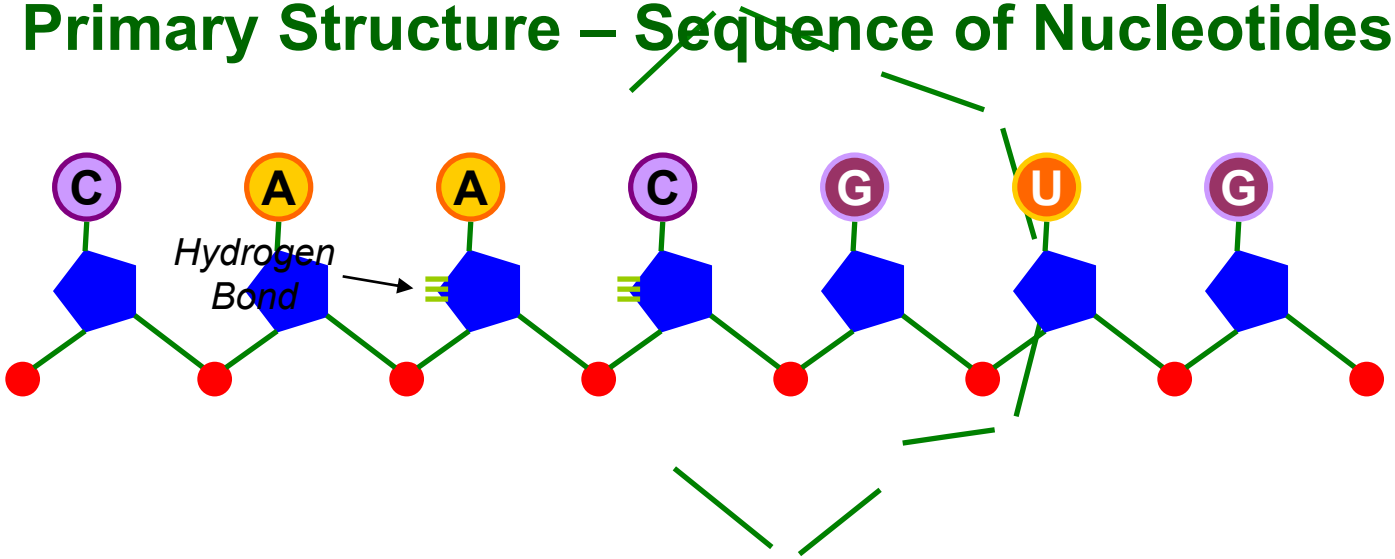
- Accumulation of Data
  - Bioinformatics, Computer Networks, and etc.
  - Massive amount of data
  - Slow streaming data processing
- Computer Networks
  - Detection, Filtering, Modifications, and Routing
  - Parallel Hardware Acceleration
  - Orders of Magnitude Speedup over SW Alone
- Research Direction
  - High Performance and Low Cost HW/SW Solutions
  - Research Infrastructure for Variety of Topics
  - Dynamic Reconfiguration for Future Network Research

one more animated slide

RNA Structure

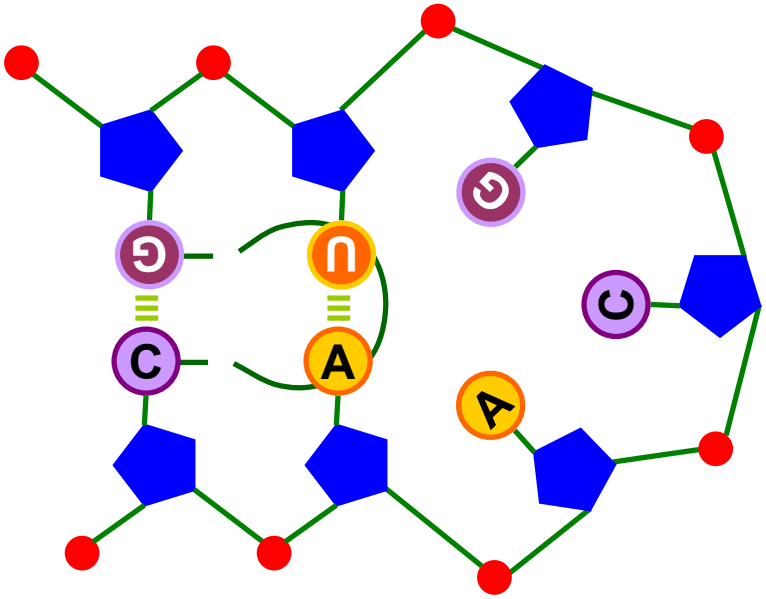
Secondary Structure – Folded 3-D Shape

Primary Structure – Sequence of Nucleotides



RNA Structure

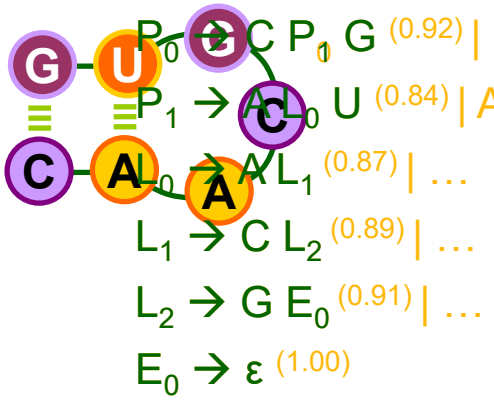
Secondary Structure – Folded 3-D Shape



RNA Structure

Secondary Structure – Folded 3-D Shape

- **Represent using Context Free Grammar**
- **Generate Stochastic CFG from Database**
- **Extend Stochastic CFG to Covariance Model**



$P_0 \rightarrow C\ P_1\ G\ ^{(0.92)} \mid C\ P_1\ A\ ^{(0.01)} \mid C\ P_1\ U\ ^{(0.02)} \mid \ldots$

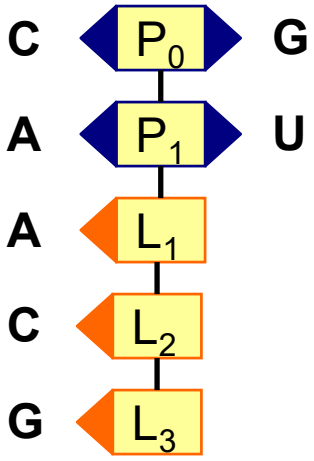
$P_1 \rightarrow A\ L_0\ U\ ^{(0.84)} \mid A\ L_0\ G\ ^{(0.03)} \mid A\ L_0\ C\ ^{(0.04)} \mid \ldots$

$L_0 \rightarrow A\ L_1\ ^{(0.87)} \mid \ldots$

$L_1 \rightarrow C\ L_2\ ^{(0.89)} \mid \ldots$

$L_2 \rightarrow G\ E_0\ ^{(0.91)} \mid \ldots$

$E_0 \rightarrow \varepsilon\ ^{(1.00)}$



Covariance model allows parsing of sequences with insertions and deletions