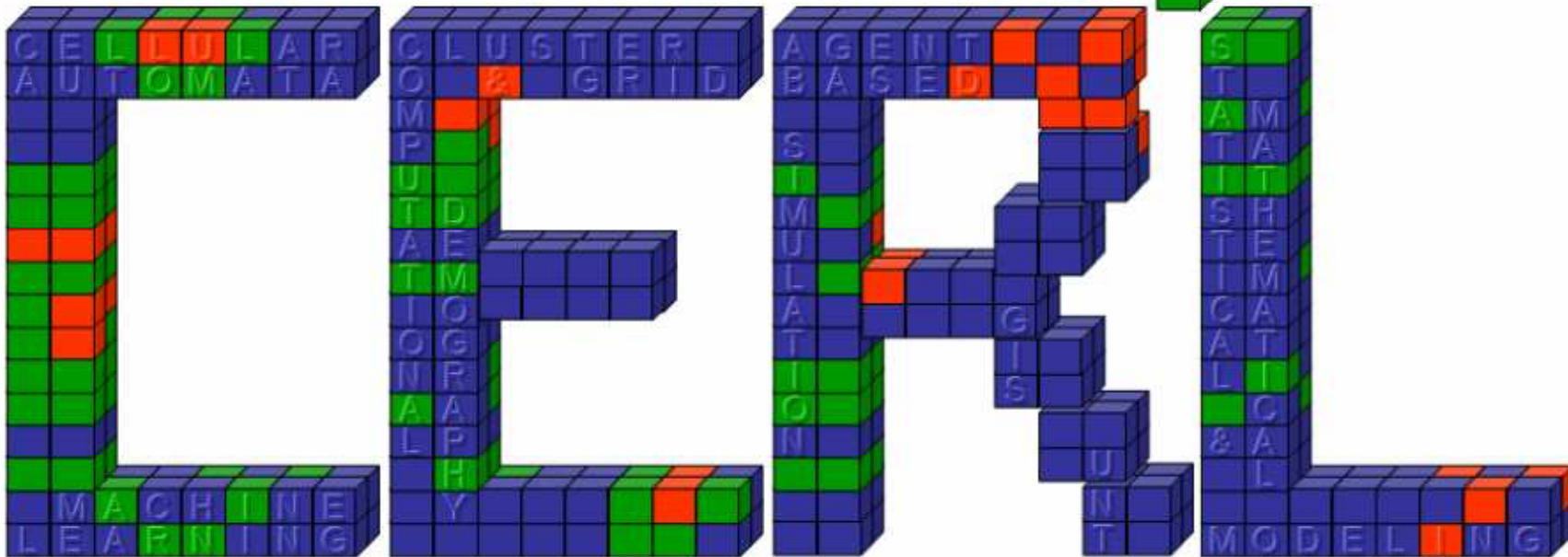


# CerIML Proposal

*CERL Standard Data Format*



**Computational**

**Epidemiology**

**Research**

**Lab**

Presented by Brandon Parker

# The Proposed Choices

- 1) Comma Separated Values file (CSV)
- 2) Binary File Format
- 3) XML (w/ DTD,XSL)

# 1.) CSV

## Benefits:

- Easy to read & write
- Human Editable
- Small storage footprint
- Easy import to SAS, SPSS, and Excel

## Drawbacks:

- Must parse carefully/manually
- Static design
- Manual conversion necessary

## 2.) Binary File

### Benefits:

- Fast to read & write
- Smallest storage footprint

### Drawbacks:

- Must parse carefully/manually
- Static design
- Not human readable
- Non-trivial manual conversion necessary

## 3.) XML

### Benefits:

- Parsers available (SAX & DOM)
- Fast non-linear parsing
- Human editable
- Universally portable
- Dynamically expandable w/ backwards compatibility
- Easy rendering (XSL)
- Validation and templates (DTD)

### Drawbacks:

- Must use parser for program input
- Large storage footprint

# What is XML?

- Document Type Definitions (DTD) or XML Schema (XSD)
- Extensible Markup Language (XML)
- Extensible Stylesheet Language (XSL)

# What is XML?

- XHTML – the new HTML standard
- OpenGIS – GIS data in XML format
- SMIL – Sync. Multimedia Integration Lang
- RSS – RDF Site Summary
- MusicXML – music notation (i.e. Finale)
- MathML – Mathematical Markup Language
- CML – Chemical Markup Language

# Example XML

```
<?xml version = "1.0">
```

```
<Document-level-tag>
```

```
  <level-1-tag attribute="VALUE">
```

```
    <level-2-tag>
```

```
      <level-3-tag>PCDATA</level-3-tag>
```

```
      <level-3-tag>PCDATA</level-3-tag>
```

```
      <single-tag attr1="VAL1" attr2="val2" />
```

```
    </level-2-tag>
```

```
  </level-1-tag>
```

```
</Document-level-tag>
```

# Example DTD

```
<!ELEMENT document-level-tag (level-1-tag)>
```

```
<!ELEMENT level-1-tag (level-2-tag)>
```

```
<!ATTLIST level-1-tag attribute type default>
```

```
<!ELEMENT level-2-tag (level-3-tag*, single-tag)>
```

```
<!ELEMENT level-3-tag #PCDATA>
```

```
<!ELEMENT single-tag EMPTY>
```

```
<!ATTLIST single-tag
```

```
    attr1          CDATA "foo"
```

```
    attr2          CDATA "bar"
```

```
>
```

# Proposed CerlML - Demo

<http://www.cerl.unt.edu/~bparker/xml>

# XML Benefits

- Import/export to many databases (DB2, Oracle, Informix, MySQL)
- XML → CVS conversion simple
- Defined standard
- Easy validation with DTD or XML Schema
- Expandable to encompass OpenGIS, disease builder, binary objects (in UUE encoding), etc
- Quick, universally readable method for sending information
- Cross-simulator/visualizer compatible

Output: <tag>your it</tag>

```
1. int main(int argv, char *argc[]) {
2.     printf("<?xml version = \"1.0\"> \n");
3.     printf("<simulation>");
4.     printf("%s",MetaData);
5.     char MyArray[255][255];
6.     foo(MyArray);
7.     printf("<data io=\"out\" type=\"misc\">\n");
8.     printf("<size dim=\"1\" width=\"255\" /> \n ");
9.     for (int i = 0; i < 255; i++)
10.         printf("<gnarf>%s</gnarf>\n", MyArray[i]);
11.     printf("</date> \n");
12.     printf("</simulation> \n");
13. }
```

# Input: XML Parsers

## **SAX** – Simple API for XML

- Parses XML in-place
- Event driven
- Handles extremely large files
- Slower processing (File IO)
- Great for 1-pass parsing

## **DOM** – Document Object Model

- Loads XML to memory in tree structure
- Faster interactive processing
- Memory Intensive

Available in these fine flavors:

C/C++

Java

JavaScript

Pascal

PHP

Perl

Python

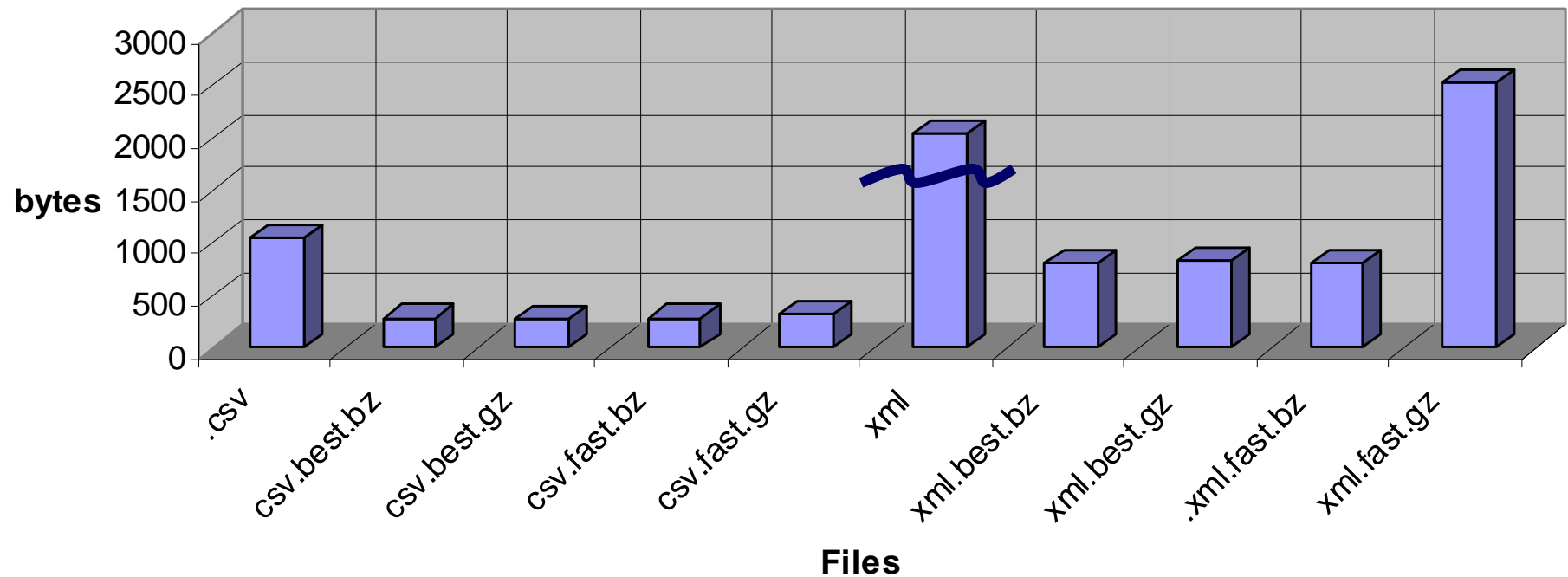
VB

VBScript

and more!

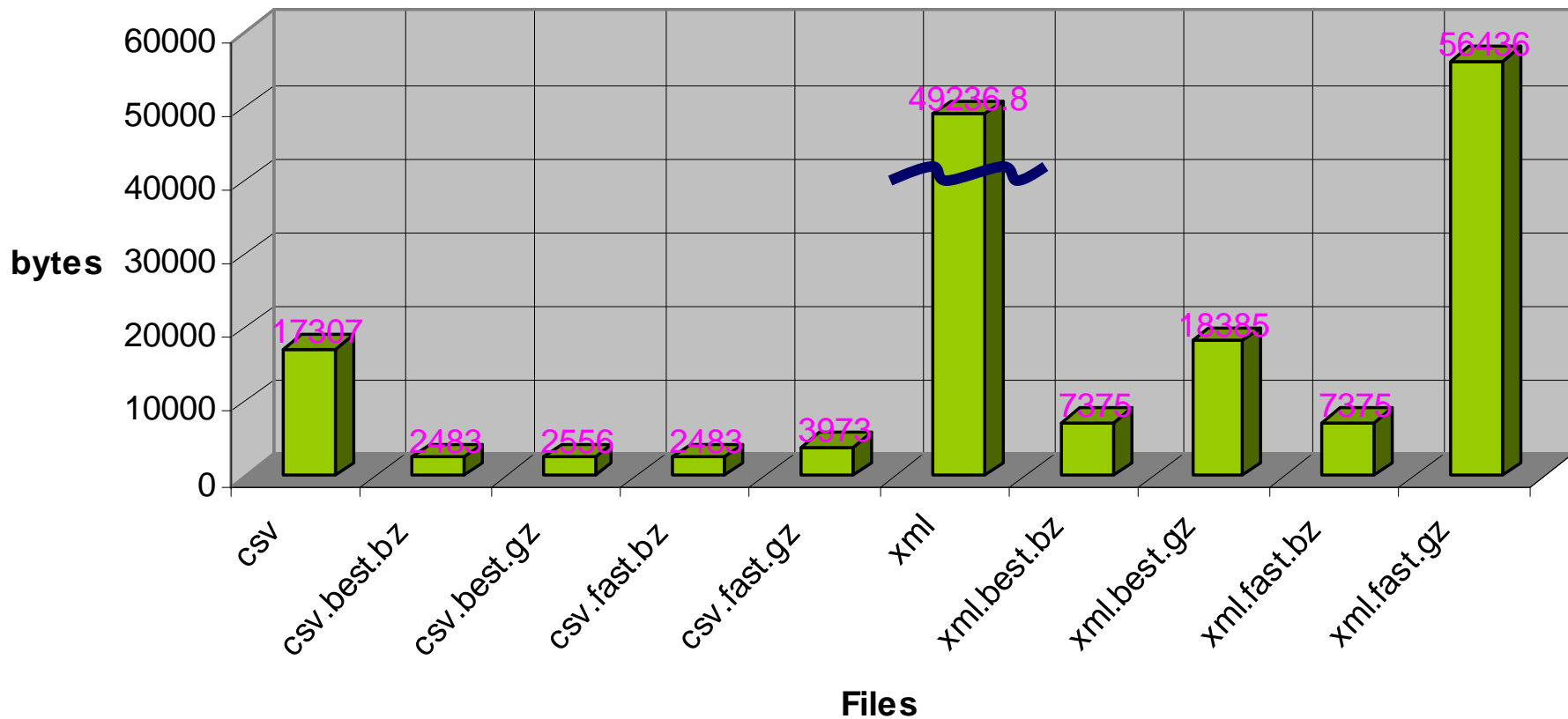
# XML vs CSV – file size

5x5 Grid File Sizes



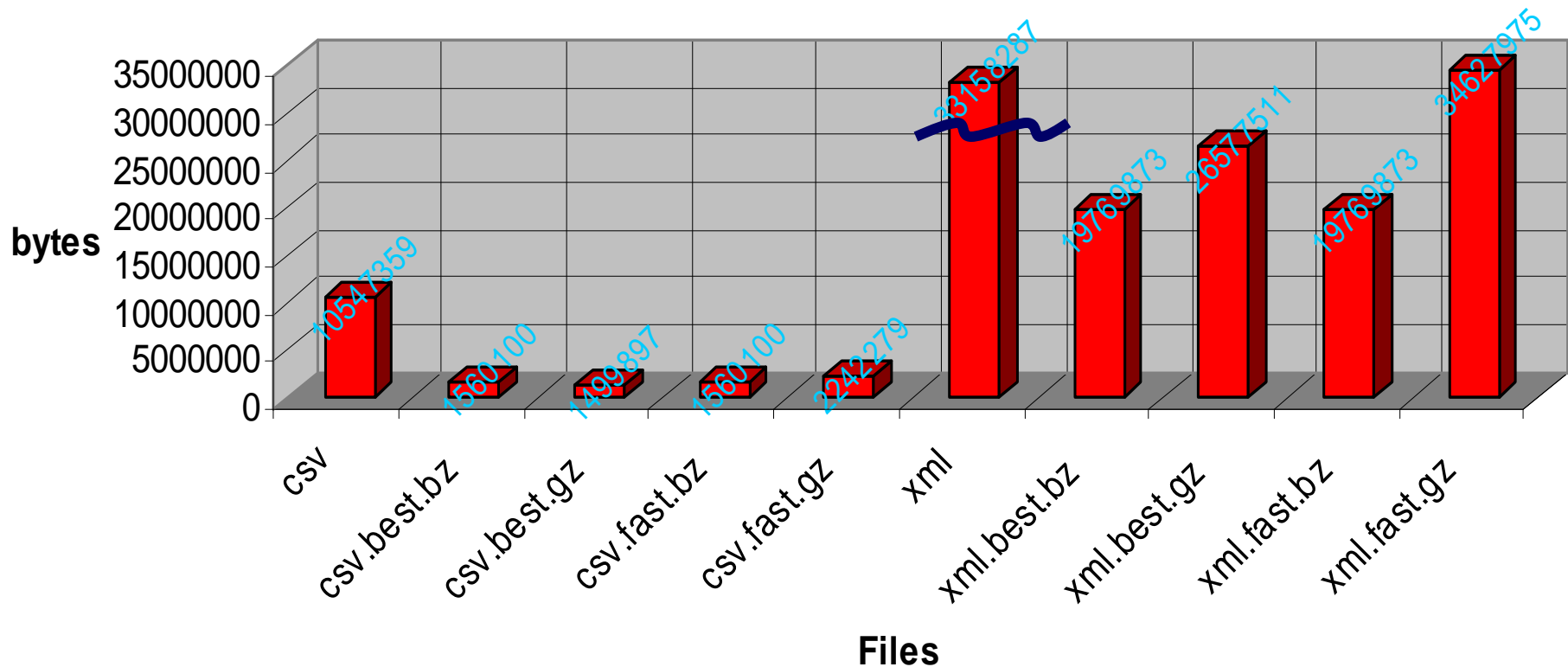
# XML vs CSV – file size

25x25 Grid File Sizes

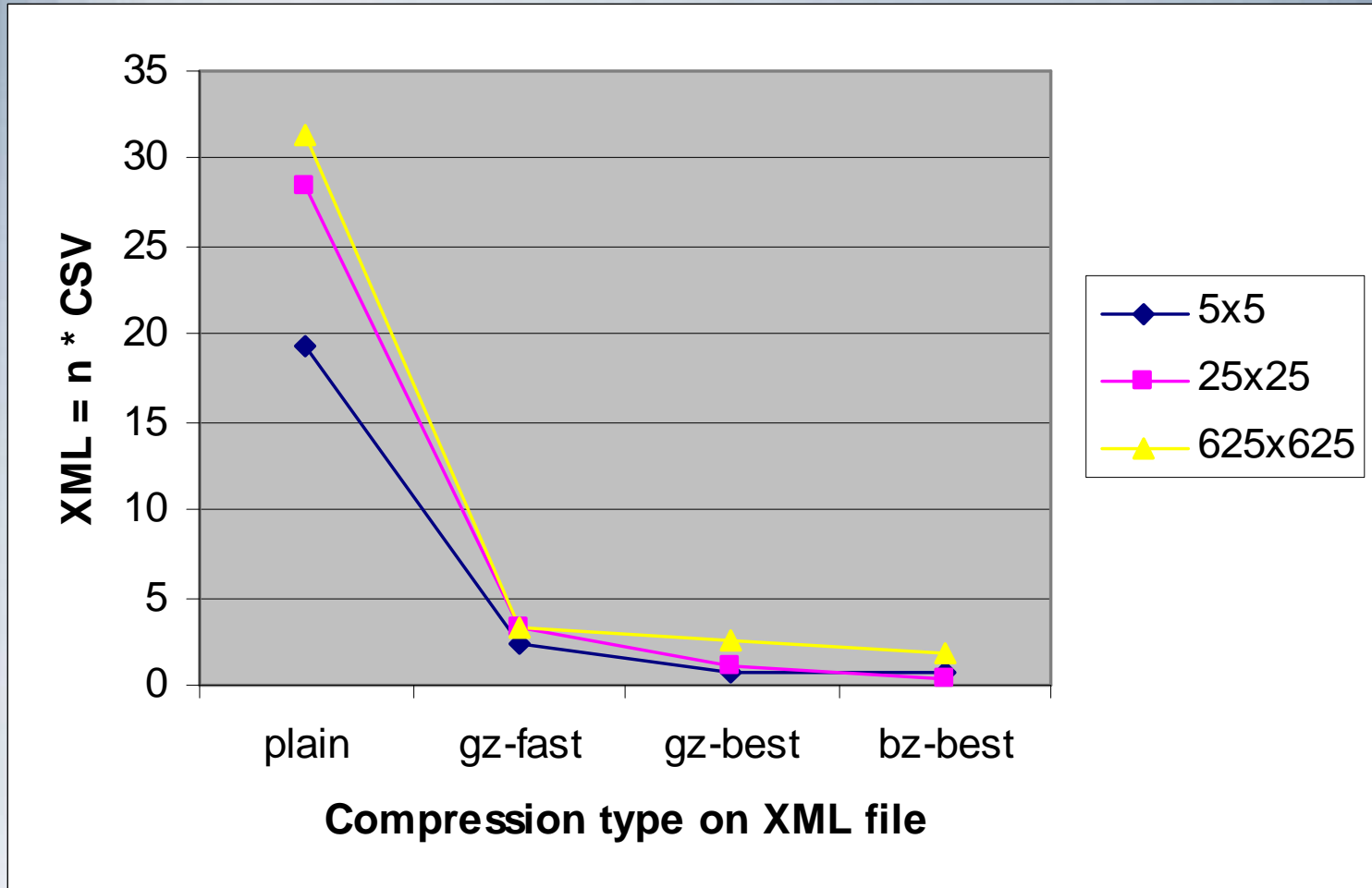


# XML vs CSV – file size

625x625

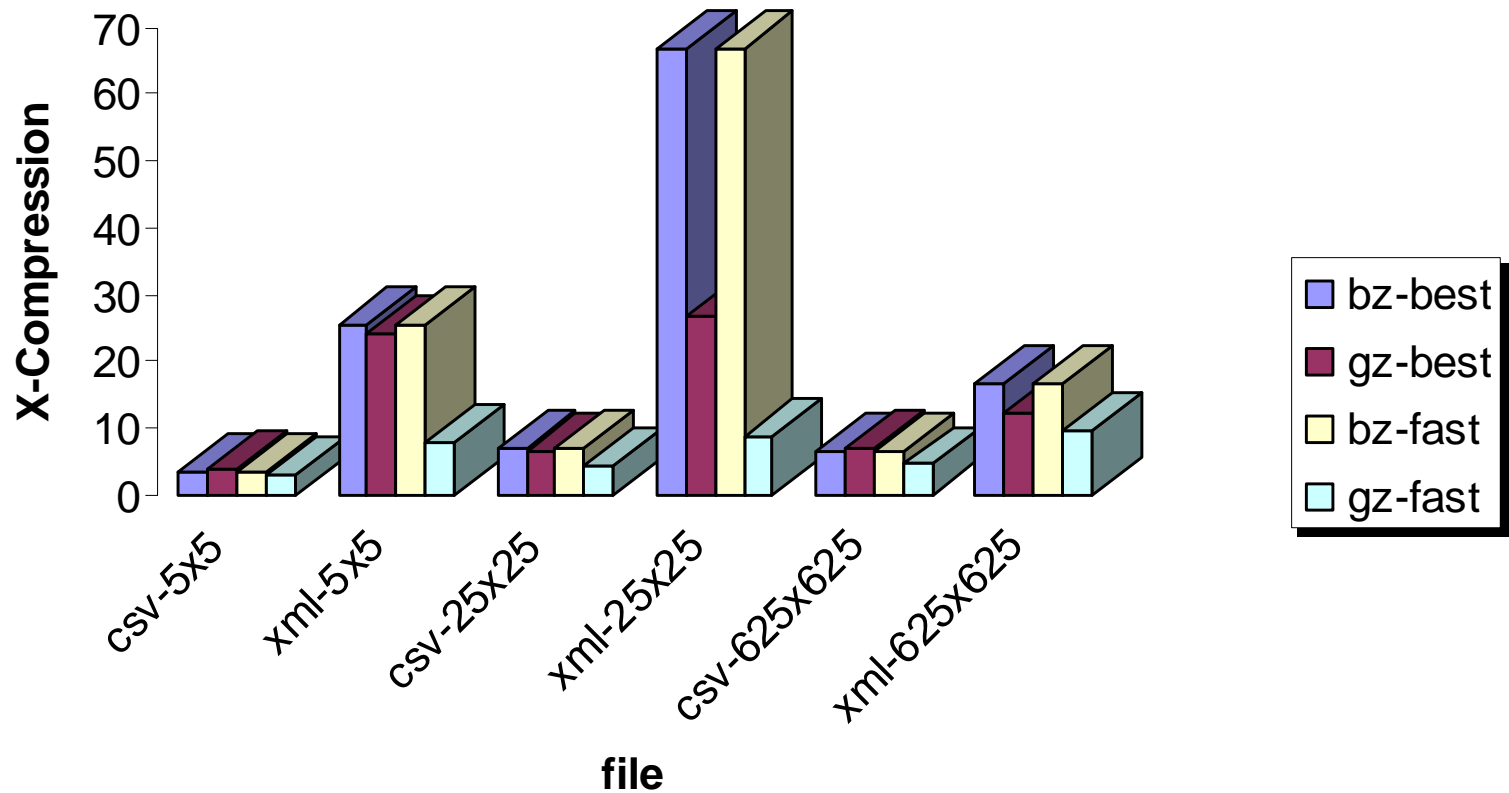


But not that bad...



# How much compression?

## Best Compression for XML and CVS



# Summary

- Size is significantly bigger!
- Not overwhelming if compressed.
- Easy to parse, write, search, render.
- Transportable.
- Expandable
- Parsers available in a language near you!

(enigma: Why is 'fast' GZip bigger?)

# Conclusion

A CERL data standard will enable better collaboration and extendibility of projects

## **Binary files**

fast, small, but inflexible

## **CSV files**

medium size and speed, somewhat flexible

## **XML files**

slower, bigger, more dynamic in content, rendering, and storage.

# For More Info...

## Web Resources:

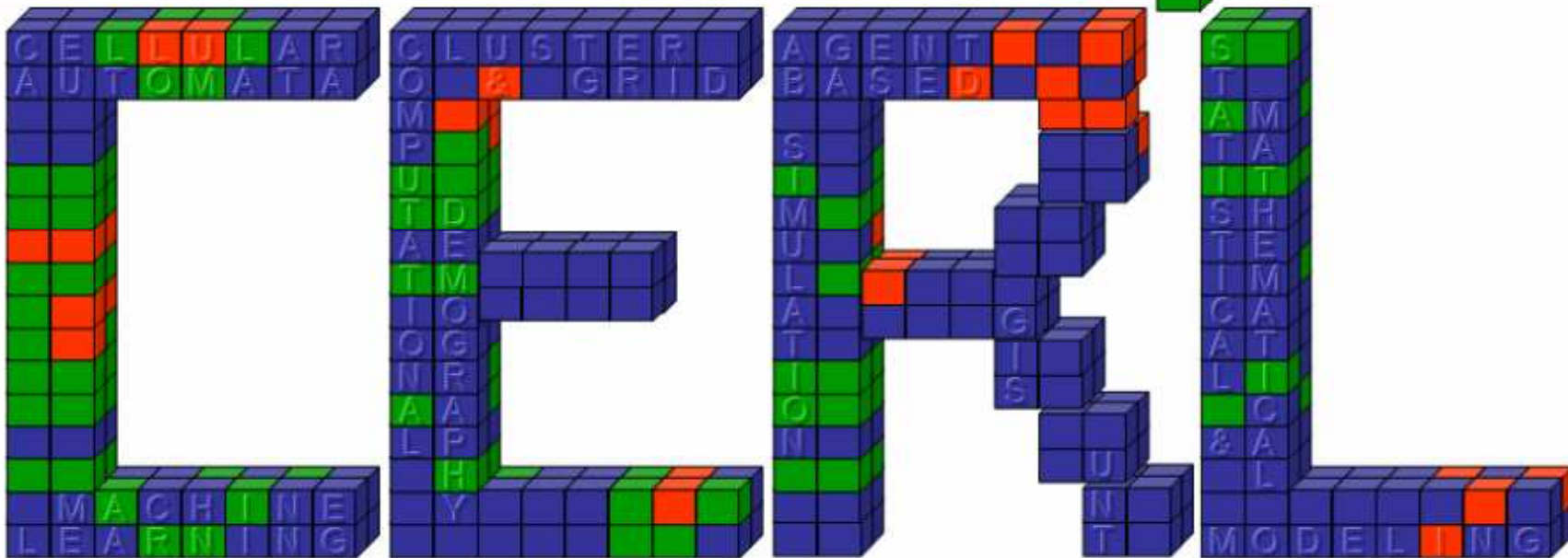
- [www.w3c.org](http://www.w3c.org)
- [www.w3schools.com](http://www.w3schools.com)
- [www.xml.com](http://www.xml.com)

## CERL Resources:

- [www.cerl.unt.edu/~bparker/xml/](http://www.cerl.unt.edu/~bparker/xml/)

# CerIML Proposal

*CERL Standard Data Format*



**Computational**

**Epidemiology**

**Research**

**Lab**

Presented by Brandon Parker