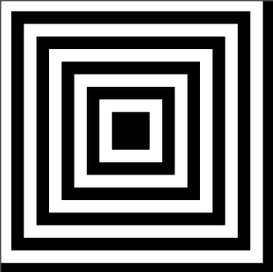
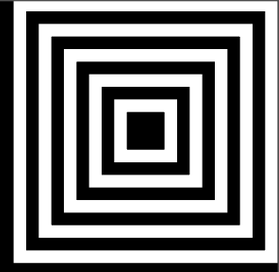
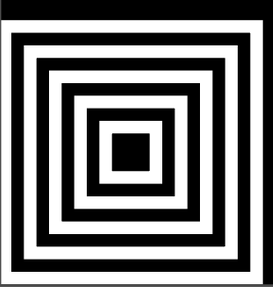
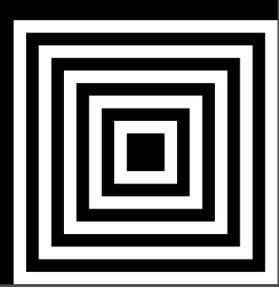



Screen Range Test

Cryptohaze Design & Architecture

Bitweasil
cryptohaze.com

How did this start?

Grad Student +
Rainbow Tables +

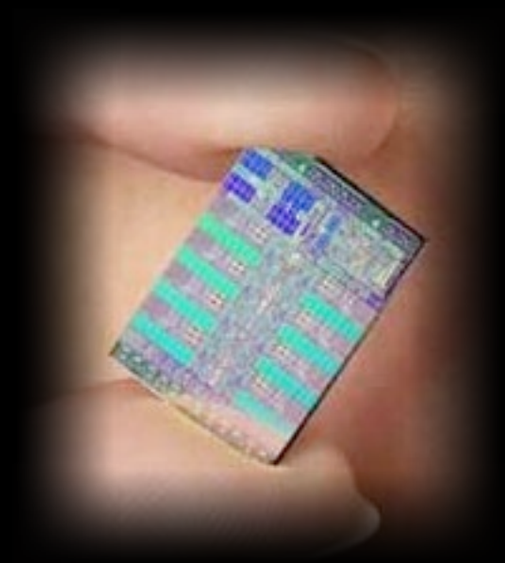
= Thesis!

Multiforcer History

- 2008: C/CUDA for rainbow table prefilter
- 2010: C++/CUDA rewrite
- 2011: Initial network support
- 2012: C++/CUDA/OpenCL/CPU rewrite
  - Goal: No Duplicate Code (or very little)

Technologies - 2007

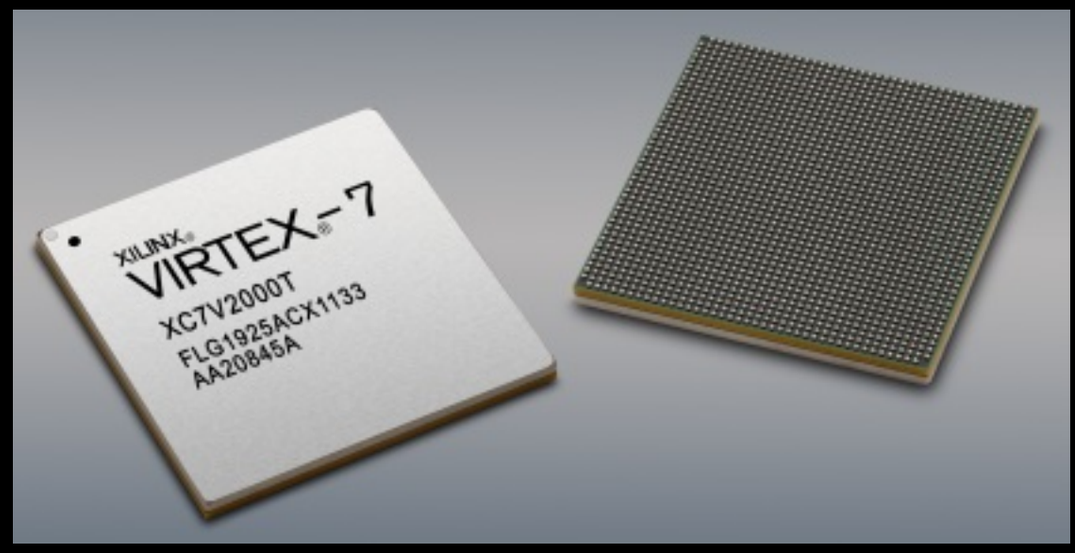
NVIDIA CUDA



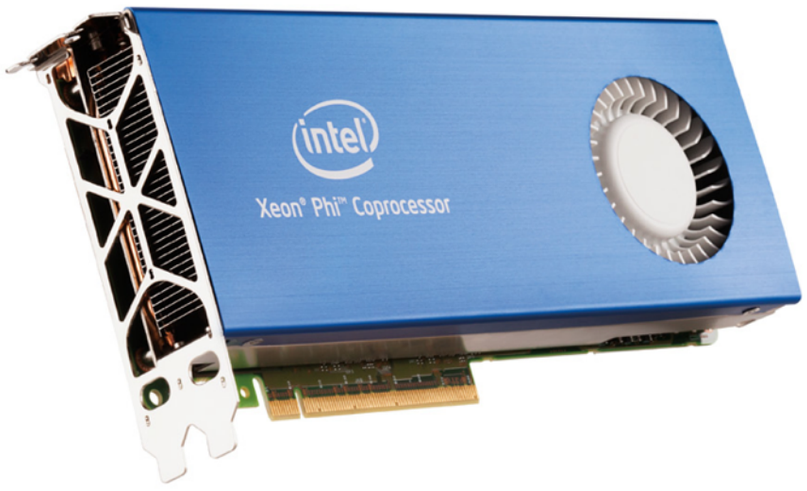
SSE/SSE2

Technologies - 2012

NVIDIA CUDA



RADEON GRAPHICS
AMD



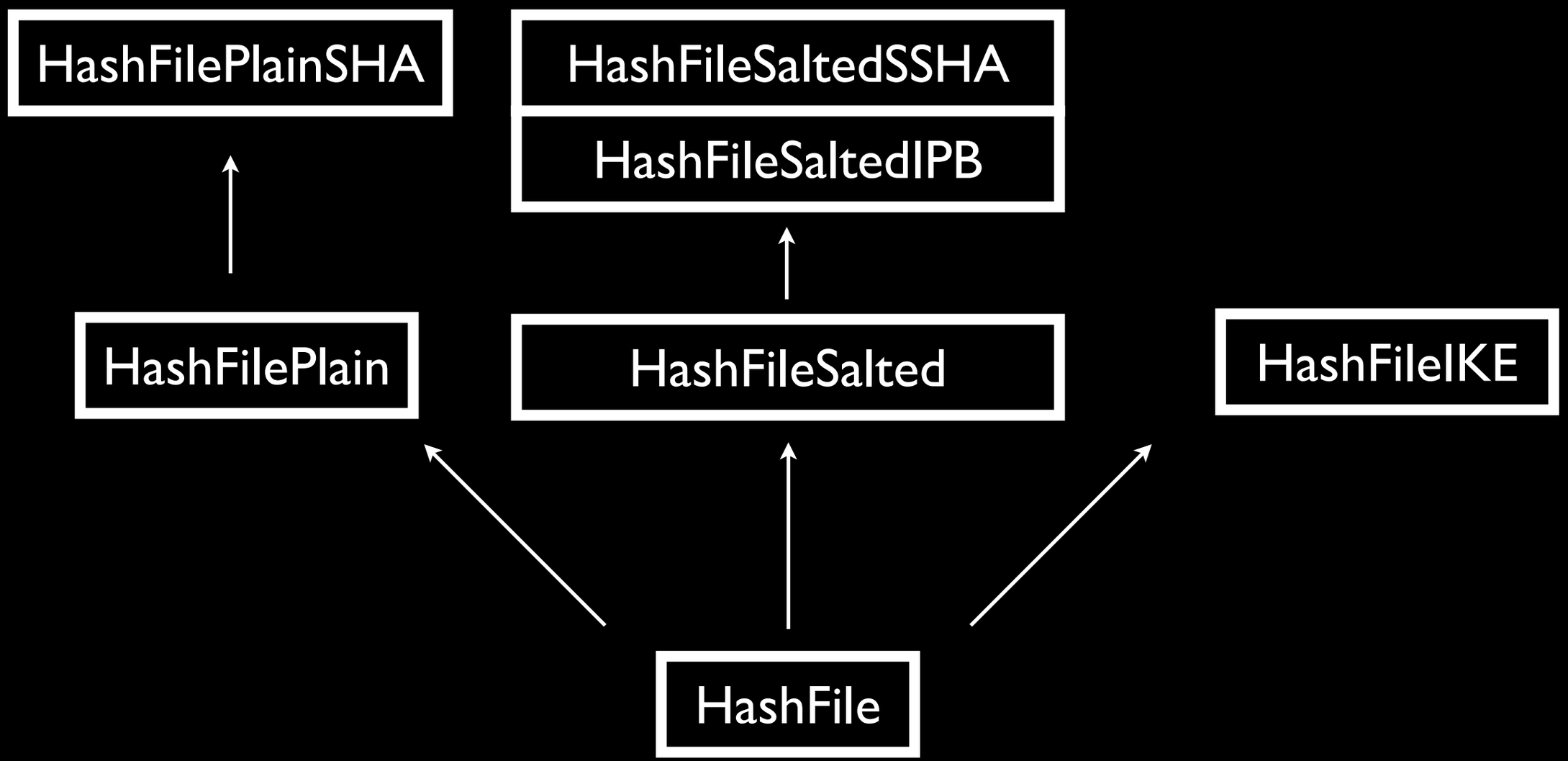
OpenCL

SSE{2,3,4}
XOP/AVX

Cryptohaze Multiforcer

- GPLv2
- Linux, Windows, OS X
- CUDA, OpenCL, CPU
- C++ Framework
- Integrated Network Support

Code Architecture



Code Architecture

{Tech}_{Hash}.{cl,cu}

Device kernel, making use of common code libraries/building blocks

Plain{Tech}_{Hash}

Kernel specific setup and transfer

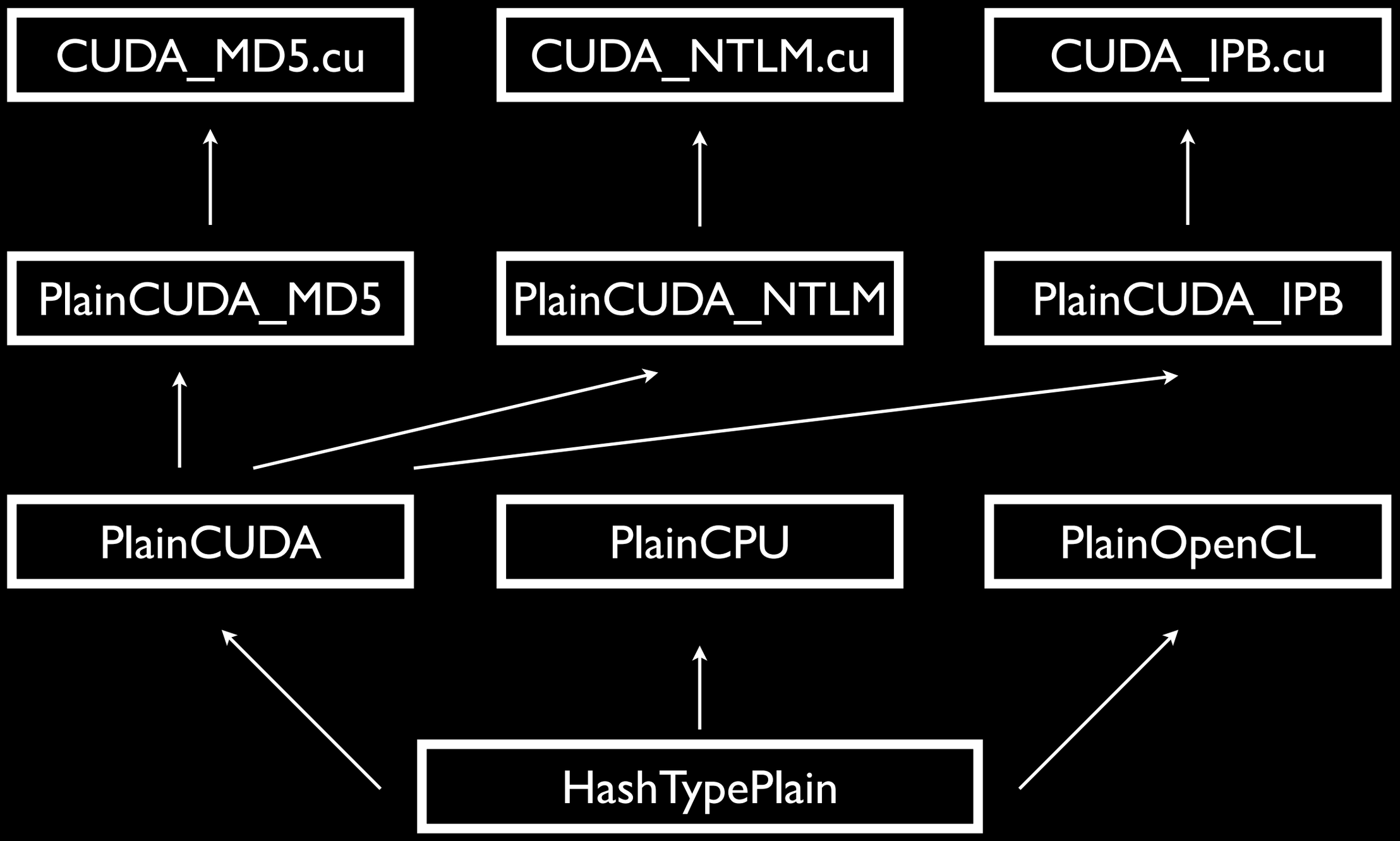
Plain{Tech}

Compilation, memory management, main data copying to device

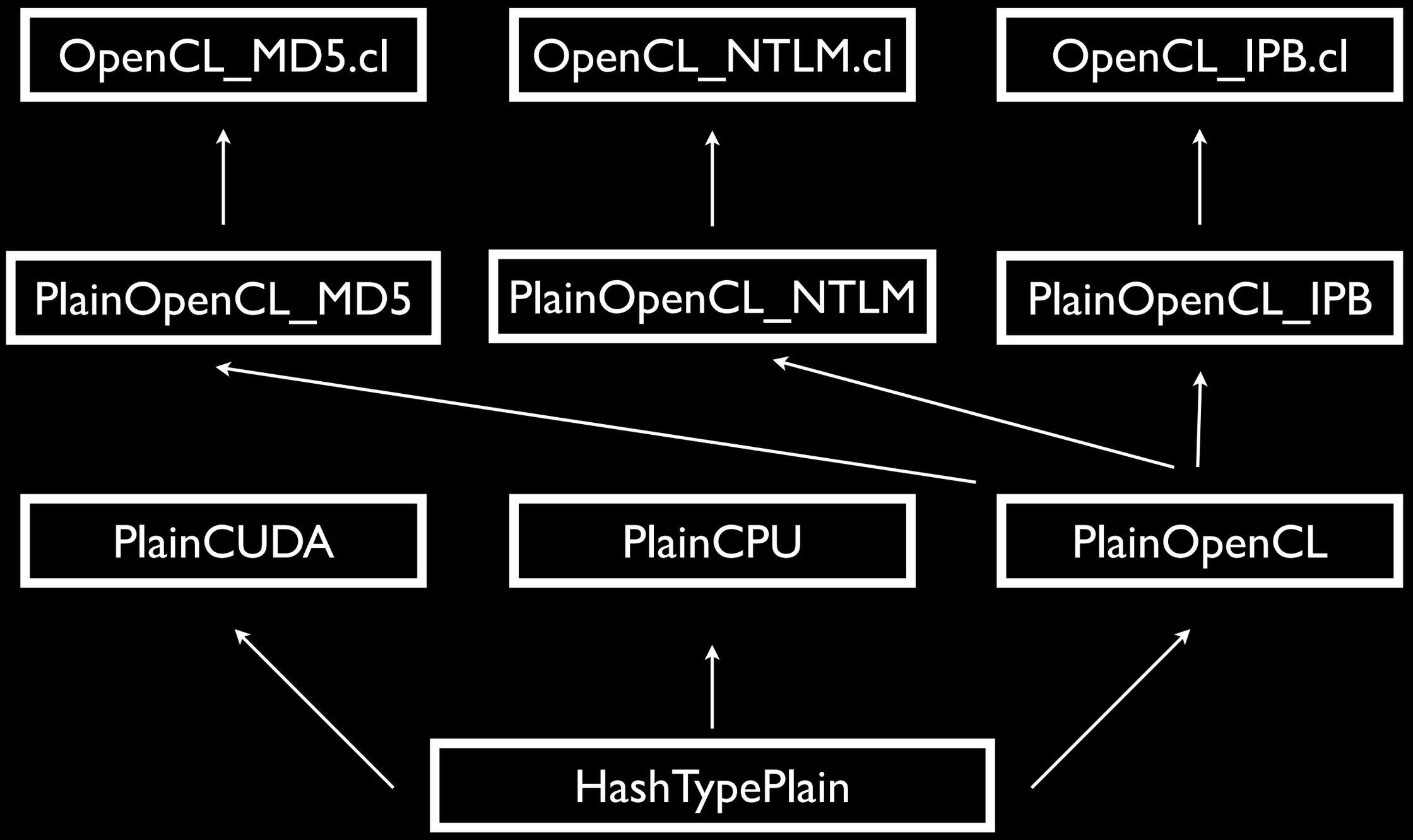
HashTypePlain

Logic, control flow, main operations

Code Architecture



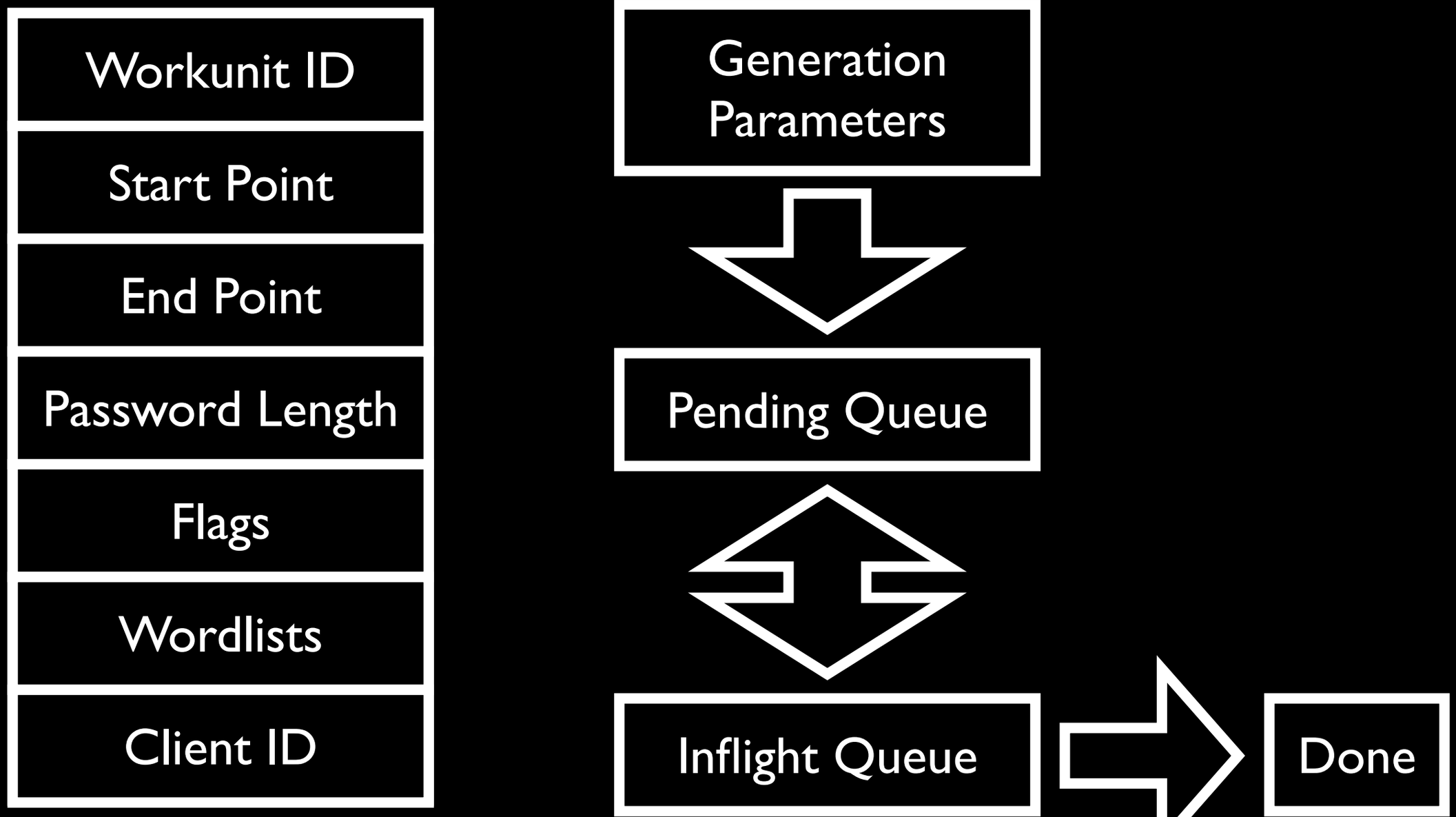
Code Architecture



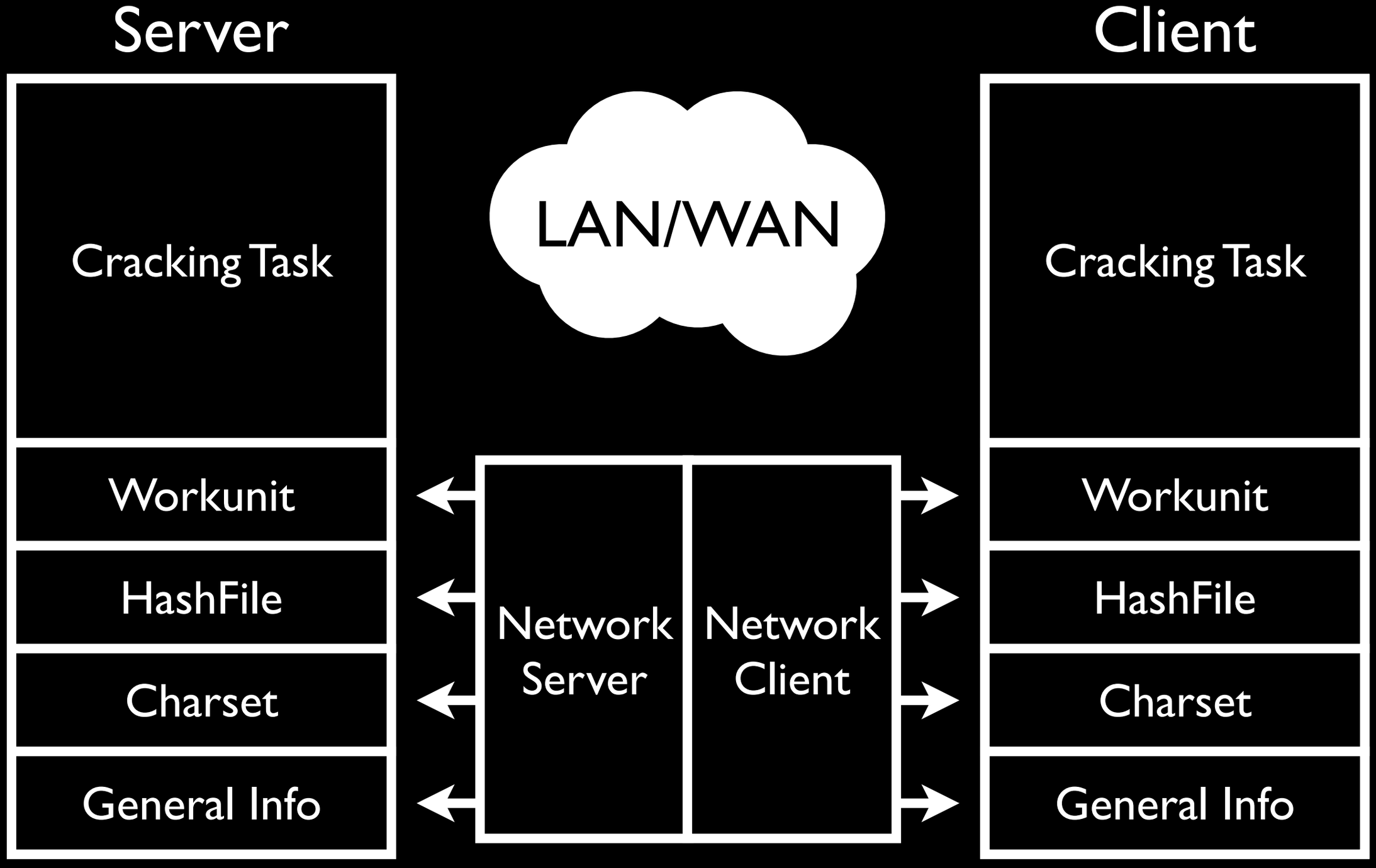
Workunits

- Workunit ID
- Start Point
- End Point
- Password Length
- Flags
- Wordlists
- Client ID

Workunits


Workunit ID
Start Point
End Point
Password Length
Flags
Wordlists
Client ID
Generation Parameters
Pending Queue
Inflight Queue
Done

Network Architecture



Import/Exports

```
void ImportHashListFromRemoteSystem(
  std::string &remoteData);

void ExportHashListToRemoteSystem(
  std::string &exportData);
```

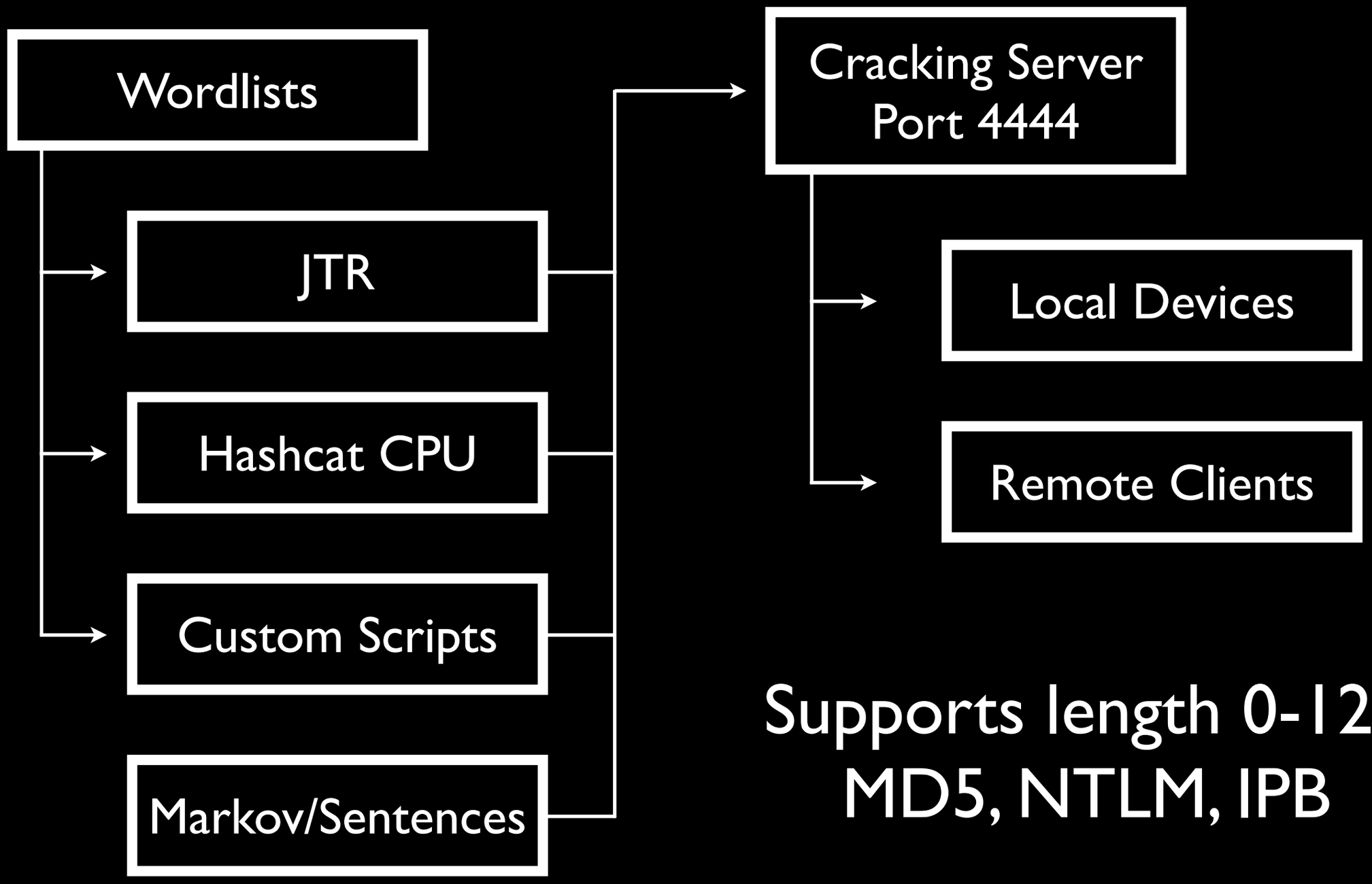
Serialization: Protocol Buffers

- Efficient, binary-packed, machine-parseable format (vs XML)
- Backwards compatible with optional fields
- Each class only talks to another instance across the network
- Network class just passes a stream of data

The Problem with Wordlist Support

- Varying maximum length supported
- Single file source, or single stdin support
- Network distributed support is hard

Cryptohaze Wordlists



Supports length 0-127
MD5, NTLM, IPB

WebTables

"Tables in the Cloud"

GPU

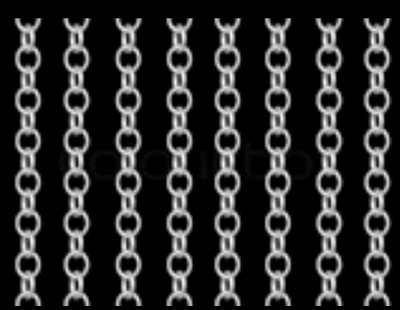
Net

deadbeef
d00df00d

Candidate
Hashes

GPU

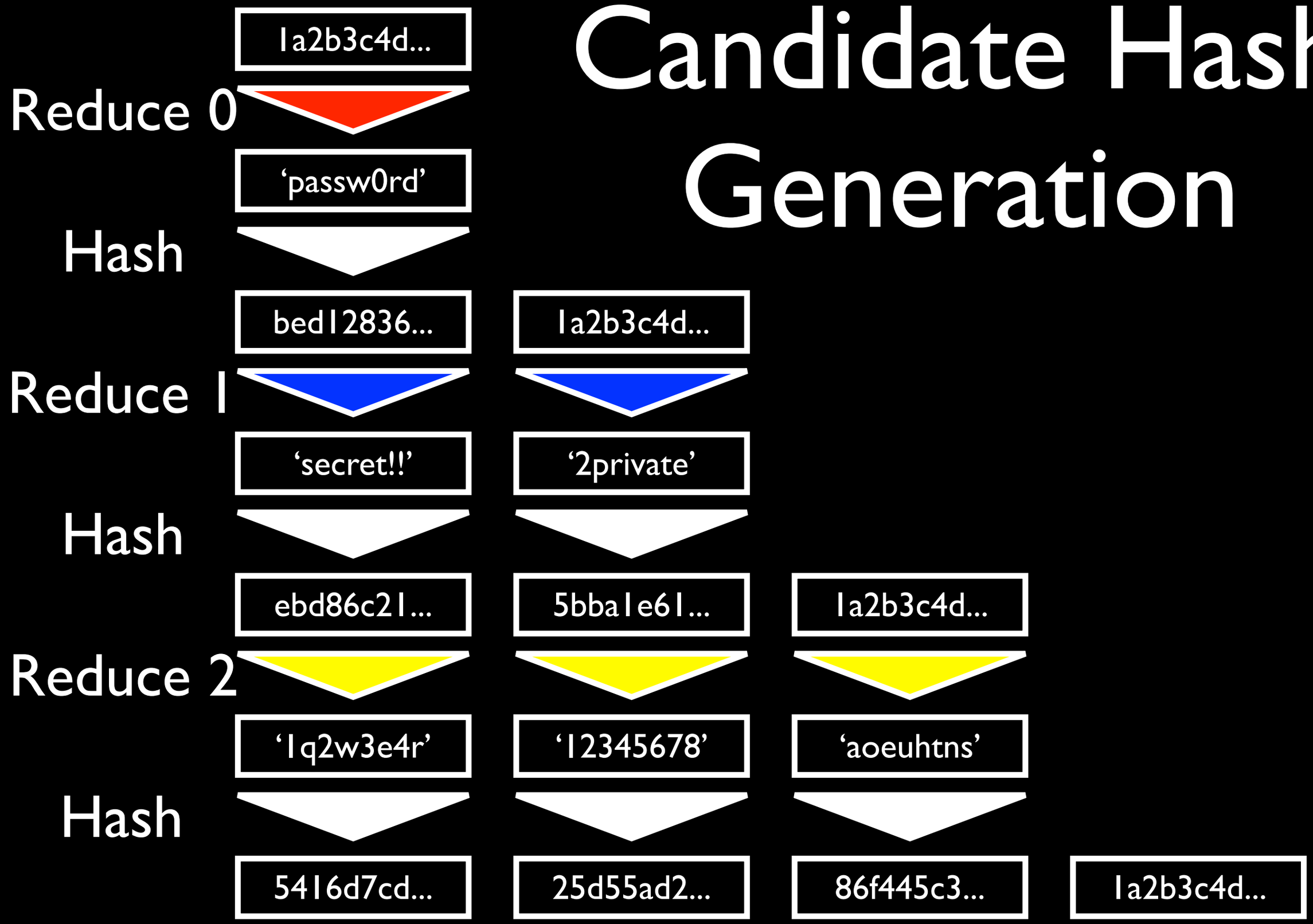
p@s5w0rd



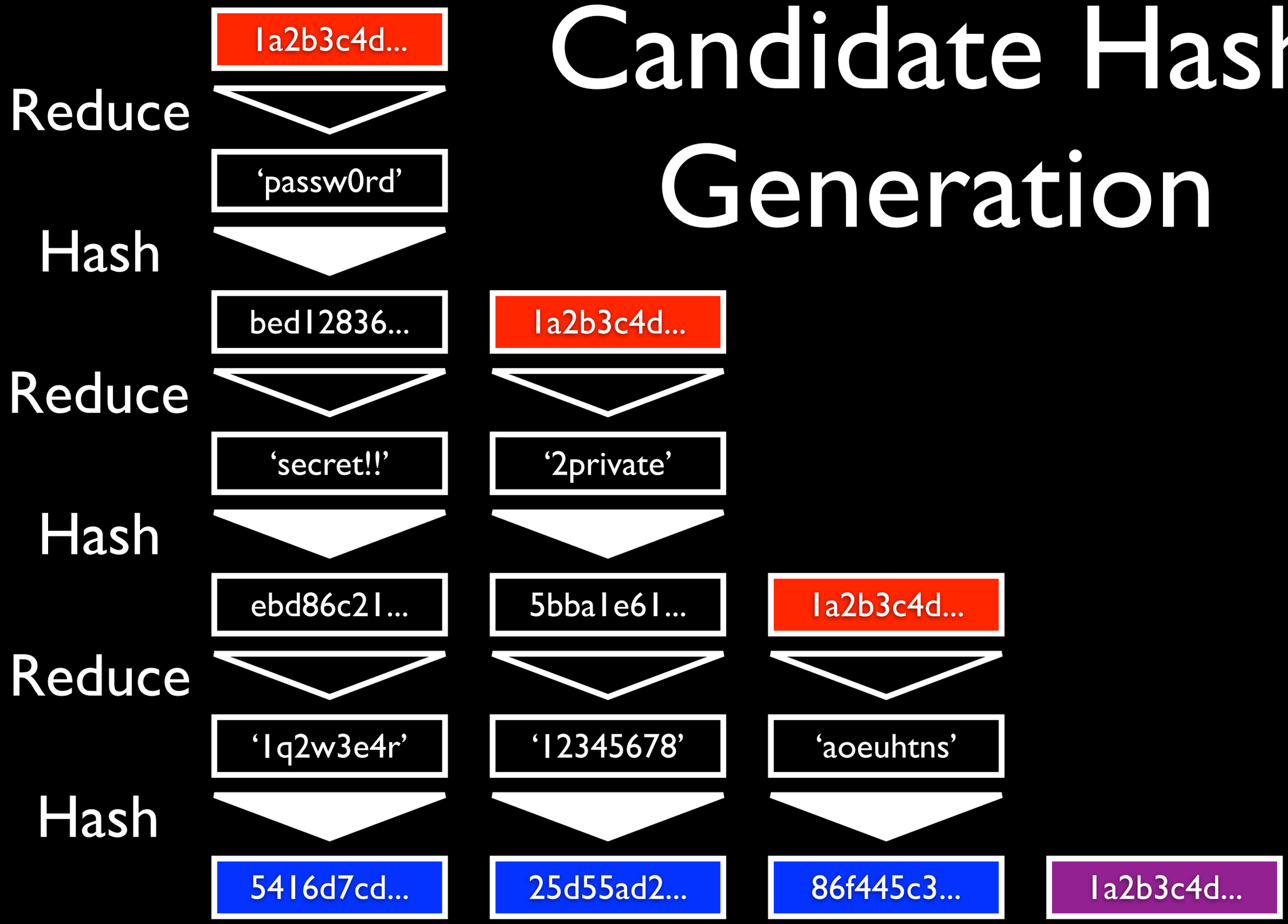
Net

Chains to Regen

Candidate Hash Generation



Candidate Hash Generation



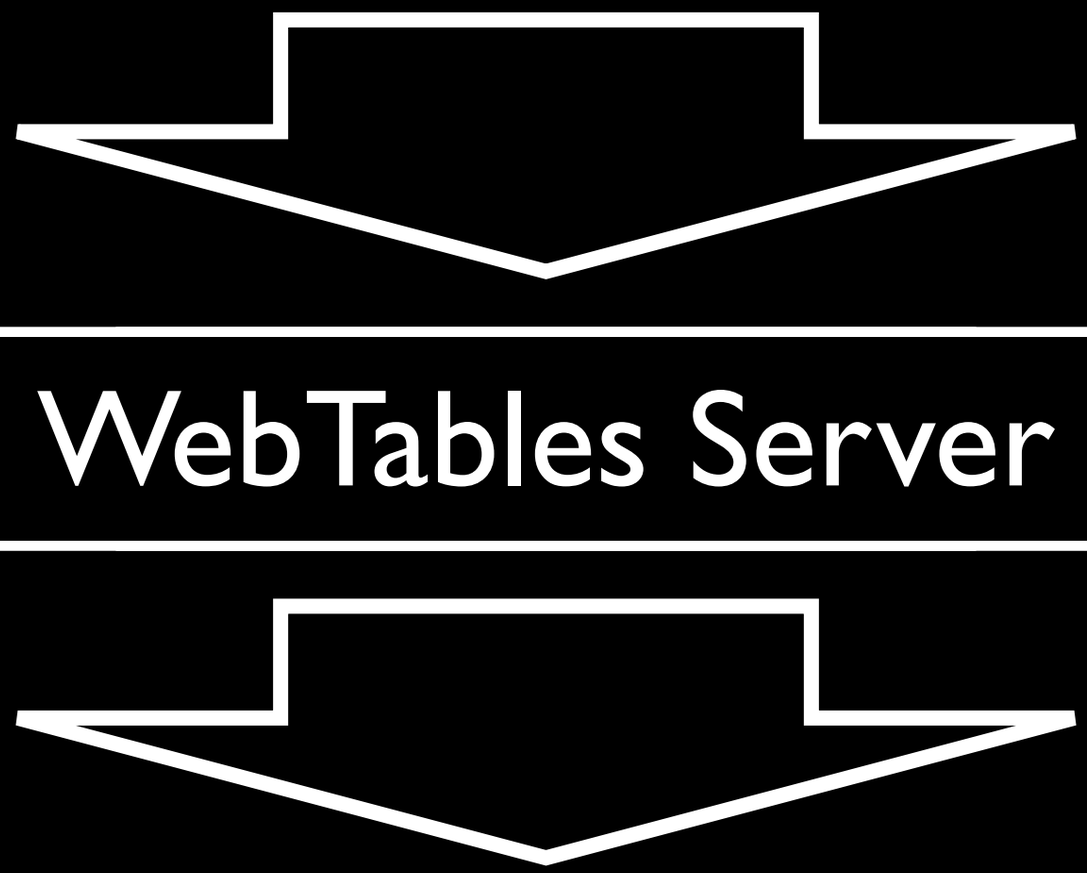
Chain Search

1a2b3c4d...

25d55ad2...

5416d7cd...

86f445c3...



‘fr&hb%hi’

‘/gxE}a(5’

‘!}aXzYXe’

‘R)mq%YOj’

WebTables Search

1a2b3c4d... | 25d55ad2... | 5416d7cd... | 86f445c3...

24 bits

1a2b3c | 25d55a | 5416d7 | 86f445

HTTP POST

1a2b3c25d55a5416d786f445

WebTables Server

1a2b3c25d55a5416d786f445

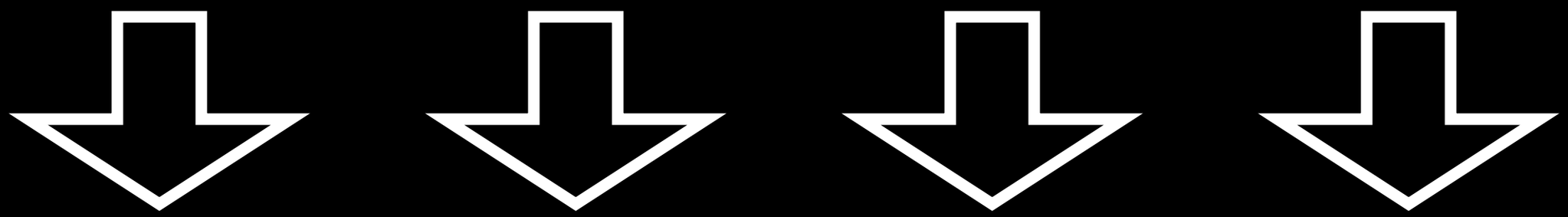
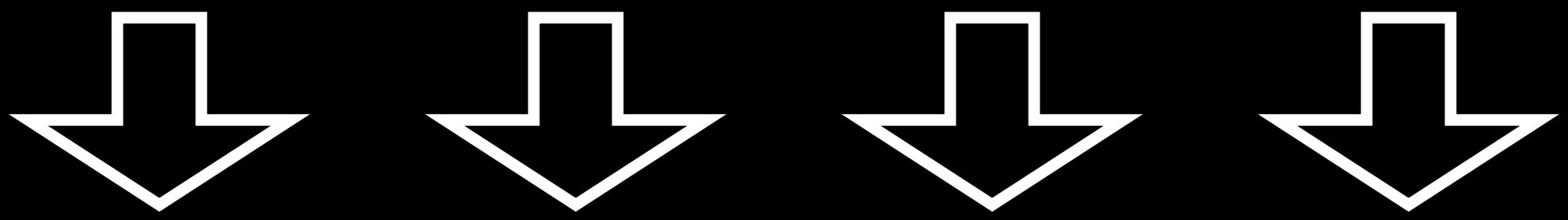


Table Server | Table Server | Table Server | Table Server



fr&hb%hi\n/gxE}a(5\n!}aXzYXe\nR)mq%YOj

WebTables

- Last 2 candidate chains skipped normally, user-selectable count
- 0.001% reduction in efficiency (with 200k chain length)
- Flexible backend - transparent to end user
- Zero table download!

Questions?

- cryptohaze.com
- webtables.cryptohaze.com
- bitweasil@cryptohaze.com
- #cryptohaze on irc.freenode.net